

JP06125454 A
PICTURE PROCESSOR
CANON INC

Abstract:

PURPOSE: To efficiently transmit picture information including fitted picture information. **CONSTITUTION:** When a page description language 200 including an instruction which fits picture information (fitting picture information 201) stored as a file subjected to non-reversible compression or the like is interpreted and expanded, it is expanded into picture information (picture information to be fitted into) in a form where fitting picture information is not fitted, and fitting picture information 201 and picture information to be fitted into are transmitted to a printer separately from each other. An instruction or the like, which indicates the fitting position of fitting picture information, in the page description language is interpreted to prepare bit map information indicating the fitting position. Bit map information and picture information to be fitted into are compressed by reversible encoding if necessary.

Inventor(s):

NAKAYAMA TADAYOSHI
MIYAKE NOBUTAKA
KONNO YUJI
NAGASHIMA YOSHITAKE
FUKUDA HIROMI

Application No. 05188399 JP05188399 JP, **Filed** 19930729, **A1 Published** 19940506

Original IPC(1-7): H04N001387

B41J00530 B41J02100 G06F01566 G06F01572 H04N00141 H04N001411

Priority:

JP 04231982 19920831

Patents Citing This One No US, EP, or WO patent/search reports have cited this patent.

(19)日本国特許庁(J P)

(12) 公開特許公報(A)

(11)特許出願公開番号

特開平6-125454

(43)公開日 平成6年(1994)5月6日

(51)Int.Cl. ⁵	識別記号	庁内整理番号	F I	技術表示箇所
H 0 4 N 1/387		4226-5C		
B 4 1 J 5/30		Z 8703-2C		
21/00		Z 8703-2C		
G 0 6 F 15/66	3 3 0	A 8420-5L		
15/72		G 9192-5L		

審査請求 未請求 請求項の数20(全 29 頁) 最終頁に続く

(21)出願番号	特願平5-188399	(71)出願人	000001007 キャノン株式会社 東京都大田区下丸子3丁目30番2号
(22)出願日	平成5年(1993)7月29日	(72)発明者	中山 忠義 東京都大田区下丸子3丁目30番2号キャ ン株式会社内
(31)優先権主張番号	特願平4-231982	(72)発明者	三宅 信孝 東京都大田区下丸子3丁目30番2号キャ ン株式会社内
(32)優先日	平4(1992)8月31日	(72)発明者	今野 裕司 東京都大田区下丸子3丁目30番2号キャ ン株式会社内
(33)優先権主張国	日本(J P)	(74)代理人	弁理士 丸島 儀一

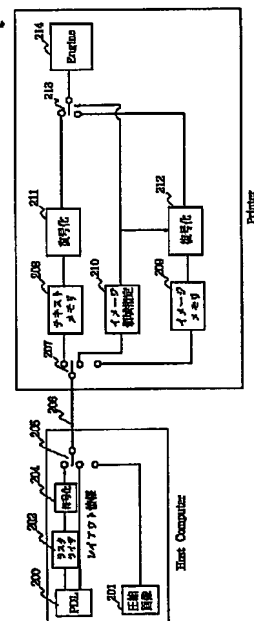
最終頁に続く

(54)【発明の名称】 画像処理装置

(57)【要約】

【目的】 効率良くはめ込み画情報を含む画情報を伝送する。

【構成】 不可逆圧縮等されたファイルとして記憶等された画情報(はめ込み画情報201)をはめ込む命令を含むページ記述言語200を解釈、展開するときに上記はめ込み画情報がはめ込まれていない形態の画情報(被はめ込み画情報)に展開202し、はめ込み画情報201と、被はめ込み画情報とを別個にプリンタに伝送する。またページ記述言語のはめ込み画情報のはめ込み位置を示す命令等を解釈してはめ込み位置を示すビットマップ情報を作成する。また必要に応じてビットマップ情報、被はめ込み画情報を可逆符号化で圧縮する。



【特許請求の範囲】

【請求項 1】 編集に用いられる画像情報と該画像情報を用いて編集を行うための領域情報を出力する手段と、該出力手段により出力された画像情報と領域情報とを伝送する伝送手段とを有し、

前記出力手段は、画像毎に展開された領域情報を圧縮した形態で出力することを特徴とする画像処理装置。

【請求項 2】 前記画像情報は圧縮されたイメージデータであることを特徴とする請求項 1 記載の画像処理装置。

【請求項 3】 前記画像情報と領域情報は、同一の方法で符号化されることを特徴とする請求項 2 記載の画像処理装置。

【請求項 4】 前記画像情報は自然画像と線画像とに分離されていることを特徴とする請求項 1 記載の画像処理装置。

【請求項 5】 前記領域情報は複数領域の情報を含むことを特徴とする請求項 1 記載の画像処理装置。

【請求項 6】 前記画像情報はページ記述言語により作成されたものであることを特徴とする請求項 1 記載の画像処理装置。

【請求項 7】 自然画像を表す第 1 の画像情報と、線画像を表す第 2 の画像情報と前記第 1、第 2 の画像情報を用いて編集するための領域情報とを発生する発生手段と、前記発生手段により発生した前記第 2 の画像情報及び領域情報を合成した後に符号化する符号化手段とを有することを特徴とする画像処理装置。

【請求項 8】 画像情報を処理するためのホストコンピュータと、該ホストコンピュータにより処理された画像情報を可視画像形成するプリンタとから構成され、前記ホストコンピュータは、自然画像を表す画素毎の画像情報を非可逆圧縮により圧縮し、線画像を表す画素毎の画像情報を可逆圧縮により圧縮して、前記プリンタに伝送することを特徴とする画像処理装置。

【請求項 9】 ページ記述言語により記述された画情報を解釈、展開してプリンタに送出する画像処理方法であって、前記ページ記述言語により、はめ込まれるべきはめ込み画情報を記憶する手段から画像展開のために前記はめ込み画情報を読み出すことなく、ページ記述言語で記述された画情報を前記はめ込み画情報がはめ込まれていない形態に展開し、前記はめ込み画情報がはめ込まれていない形態の画情報と、前記はめ込み画情報とを別個にプリンタに送出することを特徴とする画像処理方法。

【請求項 10】 請求項 9 に記載の画像処理方法において、前記はめ込み画情報がはめ込まれていない形態の画情報は圧縮してプリンタに送出することを特徴とする画

像処理方法。

【請求項 11】 請求項 10 に記載の画像処理方法において、前記はめ込み画情報がはめ込まれていない形態の画情報を可逆符号化で圧縮することを特徴とする画像処理方法。

【請求項 12】 請求項 12 に記載の画像処理方法において、前記はめ込み画情報は非可逆符号化により圧縮されていることを特徴とする画像処理方法。

【請求項 13】 請求項 9 に記載の画像処理方法において、前記ページ記述言語により記述される、はめ込み画情報ははめ込むべき位置情報に基づいて、はめ込み画情報がはめ込まれるべき位置を表すビットマップ情報を生成してプリンタに送出することを特徴とする画像処理方法。

【請求項 14】 請求項 13 に記載の画像処理方法において、はめ込み画情報を変形、又は回転してはめ込むべく、前記ビットマップ情報を処理することを特徴とする画像処理方法。

【請求項 15】 請求項 13 に記載の画像処理方法において、前記ビットマップ情報を符号化してプリンタに送出することを特徴とする画像処理方法。

【請求項 16】 請求項 15 に記載の画像処理方法において、前記ビットマップ情報と前記はめ込み画情報がはめ込まれていない形態の画情報とを共通の符号化手段により符号化することを特徴とする画像処理方法。

【請求項 17】 請求項 13 に記載の画像処理方法であって、前記ビットマップ情報がはめ込み画情報を示している画素に対応してそのはめ込み画情報の識別情報を表わし、はめ込み画情報を示していない画素に対応して色情報を表わす第 2 ビットマップ情報をプリンタに送出することを特徴とする画像処理方法。

【請求項 18】 はめ込み画像の配置領域を示す第 1 ビットマップ情報と、前記はめ込み画像のスタート位置を示す第 2 ビットマップ情報をホストコンピュータ側で PDL 展開時に生成し、PDL 展開して出来たテキスト・グラフィック情報と前記第 1、第 2 ビットマップ情報とを共通の符号化回路で符号化し、それをホストコンピュータからプリンタへ転送し、プリンタ側では、逆の符号化を行ない、前記第 1、第 2 ビットマップ情報から、複数画像の配置情報を生成し、該複数画像のレイアウトを制御することを特徴とする画像出力システム。

【請求項 19】 前記複数画像の配置情報とは、該複数のはめ込み画像の内のどの 1 つを配置すべきかを画素単位で示す情報であることを特徴とする請求項 18 記載の画像出力装置。

【請求項 20】 前記画素単位を示す情報とは、前記複数のはめ込み画像を“1”から順に番号付けした時の、各はめ込み画像に対応した番号であることを特徴とする請求項 19 記載の画像出力装置。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、画像情報を転送するホストコンピュータや、画像情報に応じて像形成を行う画像出力装置等の画像処理装置に関するものである。

【0002】

【従来の技術】ホストコンピュータとプリンタが接続されているシステムとして考えた場合には、その画像出力方法のひとつとして、ホスト側のアプリケーション等で作成したページ記述言語（以下PDLと略す）のプログラムをプリンタに転送し、プリンタがもつコントローラでPDLをビットイメージに展開して、そのビットイメージをプリンタエンジンに送り、画像を出力する方法がある。

【0003】この方法では、例えばスキャナ等から読み込まれた自然画像をPDLのプラスチック命令等により記述される線画画像にはめ込みたい場合には、上記自然画像を表わす情報をPDLのプログラムとは別途にファイルとして転送し、そのファイルを特定するコマンド及びはめ込むべき位置を特定するコマンドを含む上記PDLのプログラムをプリンタに転送する。プリンタではかかるコマンドを解釈してはめ込み処理を行う。

【0004】他の出力方法として、PDLをホスト側でビットイメージに展開し、そのビットイメージをプリンタに転送して、プリンタ側では何も処理をしないで、画像を出力する方法がある。この方法に必要な構成として、ホスト側に十分な処理能力をもつCPU、および十分な容量のメモリ、さらに展開したビットイメージを一時的に格納するのに十分な容量のハードディスクが必要となる。またホストとプリンタ間は、大容量のビットイメージを送るのに十分な速度のI/Fが必要となる。

【0005】この方法で上記自然画像を線画画像にはめ込みたい場合には、ホストコンピュータが自然画像を表わす情報をファイルとして保持しておき、そのファイルを特定するコマンド及びはめ込むべき位置を特定するコマンドを含むPDLのプログラムを解釈して線画画像に自然画像がはめ込まれた形態でビットマップイメージを生成する。かかるビットマップイメージがプリンタに転送される。

【0006】

【発明が解決しようとする課題】しかしながら、前者の方法の欠点としては、コントローラを内蔵することにより、プリンタ全体としてのコストが高くなってしまうことが挙げられる。

【0007】また後者の方法の欠点としては、ホストコンピュータに高い処理能力が要求される点の他、極めて転送速度の高い特殊なI/Fでなければプリンタ側に1ページ分のビットイメージを格納するフレームバッファが必要になることがあり、メモリのコストが非常に高くなってしまうことが挙げられる。また、低い転送速度のI/Fでもプリンタ側にフレームバッファを設けずにイ

メージデータを転送するために、ホストコンピュータで生成したビットマップイメージを圧縮してプリンタに転送し、プリンタで伸張する方法も考えられるが、線画画像に自然画像がはめ込まれたイメージを好適に圧縮する圧縮方法は考え出されていない。

【0008】本発明はかかる従来技術の欠点を除去し、効率良く画像情報の再生を行うことのできる画像処理装置を提供することを目的とする。

【0009】

【課題を解決するための手段及び作用】上記課題を解決するため、本発明の画像処理装置は、編集に用いられる画像情報と該画像情報を用いて編集を行うための領域情報を出力する出力手段と、該出力手段により出力された画像情報と領域情報とを伝送する伝送手段とを有し、前記出力手段は、画像毎に展開された領域情報を圧縮した形態で出力することを特徴とする。

【0010】また、自然画像を表す第1の画像情報と線画画像を表す第2の画像情報と前記第1、第2の画像情報を用いて編集するための領域情報とを発生する発生手段と、前記発生手段により発生した前記第2の画像情報及び領域情報を合成した後に符号化する符号化手段とを有することを特徴とする。

【0011】更に、画像情報を処理するためのホストコンピュータと該ホストコンピュータにより処理された画像情報を可視画像形成するプリンタから構成され、前記ホストコンピュータは、自然画像を表す画素毎の画像情報を非可逆圧縮により圧縮し、線画画像を表す画素毎の画像情報を可逆圧縮により圧縮して、前記プリンタに転送することを特徴とする。

【0012】

【実施例】

（第1の実施例）本発明の第1の実施例を図1を用いて説明する。

【0013】図1はホストコンピュータとプリンタがインターフェースケーブル206で接続されているシステムを簡単にブロック図で示したものである。今、ホスト側から文字や線画（以降テキストと呼ぶ）とイメージリダ等により原稿を読み取って得られた自然画（以降イメージと呼ぶ）が混在している画像をプリンタから出力することを考える。テキストはPDLプログラム200の記述を解釈することにより生成される。またイメージの情報は高い圧縮率の圧縮手段により圧縮されてホスト側のハードディスク内に置かれている（図2中の201）。この圧縮手段としてはJPEG（Joint Photographic Expert Group）で提案されている高能率の圧縮手段等が考えられる。

【0014】このイメージについてPDLプログラム中には、イメージをページ内のどの位置に張り込むかを示すレイアウト情報だけが記述されている。レイアウト情報とは、図2に示すように、イメージの書き出し位置

(主走査方向の座標：Xs、副走査方向の座標：Ys)、およびイメージの大きさ(横幅：Width、縦幅：Height)で指定する。PDLプログラムはラスライザ202でビットイメージに展開される。同時にPDLプログラムを解釈してイメージのレイアウト情報が生成される。しかし、ここではテキストにイメージのはめ込み処理を行わない。ラスライザにより展開されたビットイメージ情報は、符号化部204でテキストに適した算術符号化やランレングス符号化等の符号化手段により圧縮処理が行なわれる。

【0015】このようにして得られたテキストとイメージの圧縮された情報、およびイメージのレイアウト情報がセクタ205で選択されて所定のタイミングで、プリンタに転送される。またプリンタでは所定のタイミングでセクタ207が切り替わり、テキストの情報はテキストメモリ208へ、イメージの情報はイメージメモリ209へ格納される。またレイアウト情報はイメージ領域指定部210に入力される。

【0016】圧縮されたテキストおよびイメージの情報は、復号化部211、212で復号化される。このとき図3に示すように、テキストの復号化は常に行なうが、イメージの復号化はイメージの領域でのみ行なうので、そのタイミングを制御する必要がある。この制御をするのが、イメージ領域指定部210である。

【0017】このイメージ領域指定部210を詳細に示した図を、図4に示す。図4の点線で表わされた部分が図1中のイメージ領域指定部210に相当する。ホスト側より入力されたレイアウト情報のXs、Ys、Width、Heightは、セクタ501が所定のタイミングで切り替わることによって、503、504、506、507の各レジスタにセットされる。また復号化に伴って、主走査方向カウンタ502および副走査方向カウンタ505がカウンタ動作を行なう。すなわちこのカウンタ値が復号を行なっているページ内の座標を表わす。主走査方向カウンタ502によるカウンタ値とレジスタ503にセットされているXsの値が比較器510に入力され、カウンタ値AとXsの値Bにおいて、 $A \geq B$ のとき比較器からHighの信号が出力される。また503と504のレジスタにセットされているデータを、加算器508によって加算し、その結果を比較器511に入力する。同じようにカウンタ値Aと加算結果Bが $A \geq B$ のとき出力をHighとする。さらにこの出力が否定回路515に入力される。すなわちこの否定回路515の出力信号がHighとなるのは、比較器511において $A < B$ が成り立つ時である。副走査方向においても同様に行ない、510～513の4つの比較器の出力及びそれらを反転させた信号がAND回路514に入力される。以上の処理を行なうことにより、図2において斜線で示されたイメージの部分でのみAND回路514の出力がHighとなる。すなわち図4で示されたイ

メージ領域指定部によって、イメージ領域を復号しているときにのみ状態がHighとなる信号を生成させることができ、この信号を用いて図1中のイメージの復号化部212を制御することが可能となる。さらにその信号はセクタ213に入力され、テキスト情報とイメージ情報を切り換えてプリンタエンジン214に送られる。

【0018】この構成で、テキストとイメージの混在するような画像でもそれぞれに適した圧縮手段によりそれぞれの情報量を減らすことができ、その分プリンタのフレームバッファの容量を大幅に削減することができる。

【0019】(第2の実施例)しかしながら図1の構成では、イメージ領域指定部の構成が、図4に示すように複雑になり、復号における処理の負荷も高くなってしまうことがある。また、イメージが1ページ内に複数あるときには、複数のレイアウト情報を格納するための複数のレジスタを用意するか、あるいはメモリ等に一時格納してから、随時レイアウト情報をメモリから読み出すというような処理が必要になり、さらにカウンタや比較器等も複数必要となってしまう、複雑なハード構成となってしまうことがある。

【0020】図5は本発明第2の実施例を示す要部ブロック図である。

【0021】101はPDL(ページ記述言語)を示し、文字や図形、イメージ等の文書記述命令が定まった言語形式にて記載するブロックである。ここでは、1ページの文書中に文字、図形とともに自然画像やCG(コンピュータグラフィック)等の既存のイメージファイルを合成しようとする場合には、合成する被画像をはめ込む領域を指定する記述することになる。

【0022】102はラスライザを示し、PDL部101にて作成された文書記述命令を解釈して、ドット表示信号に変換するブロックである。また、この場合はテキスト画像は2値に限定せず、1画素で階調情報を有する、いわゆる多値でもかまわない。多値の場合にはその情報を表現するのに必要な分だけ、ビットプレーンごとにビットマップを保持する構成にすれば良い。

【0023】一方、PDL部101において、画像はめ込み領域と指定された部分には、その領域を指し示すレイアウトプレーンを作成する(レイアウトプレーン作成部11)。

【0024】このレイアウトプレーンの作成が本実施例の特徴である。

【0025】図6において、レイアウトプレーンを説明する。

【0026】作成文書の一例として図2の様に、文字、図形と同一ページ内に画像ファイルがはめ込まれるものとする。

【0027】画像ファイルがはめ込みになる領域は破線で囲んだ部分である。PDLプログラム中の、この画像領域に関する記述により、図6(b)に示す様なビット

マップを作成する。これは、ラスタイザ102により展開されるビットマップと、同一解像度のものであり、画像がはめ込まれる領域の画素部にのみ“1”が置かれ、画像領域外の画素部には“0”が置かれる。

【0028】次にラスタイザ102により作成されたビットマップと、レイアウトプレーン作成部11により作成されたレイアウトプレーンは、符号化部103において、伝送路（ホストコンピュータプリンタ間）での転送時間を短くする為に、また、プリンタ側でのメモリを軽減する為に、等の目的により、符号化が施される。この符号化方式は限定しないが、算術符号化、MMR等の可逆符号化が好ましい。レイアウトプレーンは画像領域が示されているだけであるので、変化点が少なく、符号化効率は非常に大きくすることが期待できる。すなわち、レイアウトプレーンを付加した分の符号量の増加は微々たるものである。

【0029】さて、指定領域にはめ込む画像は、本実施例では、既に、異なるアプリケーションソフトにより作成され、JPEG等の圧縮フォーマットにより、ホストコンピュータ上の記憶手段に圧縮されて格納されている画像ファイルとする。指定された画像ファイルは復号せずにプリンタに送信され、第1実施例と同様、プリンタ内に内蔵したイメージメモリ106に格納される。

【0030】また、符号化手段103において、符号化された文字、線画等の情報（テキスト情報）と、レイアウト情報はプリンタ内に内蔵したテキストメモリ105に格納される。

【0031】このプリンタ内に内蔵した2種のメモリの容量は、出力する1ページ分のフルメモリ、すなわち、（プリンタの最大出力サイズの全面素数）×（1画素当りの階調数）よりも少量であるものとする。

【0032】プリンタ内ではプリンタエンジン112の出力タイミングに同期して、テキストの復号化を始める。これは、例えば、本実施例に係るプリンタがレーザービームプリンタ、LEDプリンタ、液晶プリンタ等の場合には、プリント動作を始めたらず途中で動作を停止することが困難であるために、復号化とプリンタエンジンの出力とを同時にしなくてはならない。

【0033】テキストメモリの復号化は、文字、図形情報が多値である場合においても、ビットプレーン順次ではなく、画素順次に復号したほうが好ましい。

【0034】レイアウト情報もテキスト情報の同一画素上のひとつのビット情報として扱い、実際にはテキスト情報よりも数画素分早めに復号していく方がよい。

【0035】さて、レイアウト情報を復号していき、その結果は画像復号化手段110に送信される。この画像復号化手段110は、ホストコンピュータ上で圧縮され、プリンタ内のイメージメモリに格納された圧縮画像を復号する為の手段である。

【0036】画像復号化手段110は、レイアウト情報

が“0”の時は復号動作を行わず、レイアウト情報が“1”になった時のみ復号動作を始める。

【0037】また、同じレイアウト情報はスイッチ111に送信されて、レイアウト情報が“0”の時には端子Aに接続され、テキスト情報の出力を選択し、“1”になった時には端子Bに接続されて、復号された画像情報の出力を選択する。

【0038】このように、レイアウト情報をビットマップにて全面素分保持することにより、テキスト情報と画像情報との切り換えが非常に容易なものとなる。

【0039】次に、本実施例による他の効果を説明する。

【0040】図7（A）は前述した説明と同様、ホストコンピュータ上で作成する文書を表わしている。この作成文書も文字、図形と同一ページ内に、他のアプリケーションソフトにて作成された既存の画像ファイルがはめ込まれるとする。この画像ファイルは図7（B）に示した様に、もともとは矩形画像であり、JPEG等の圧縮方式により符号化されたまま、ホストコンピュータ上の記憶手段に格納されている。

【0041】いま、作成する文書上で、図7（A）中の破線で示しているように画像そのものに変形を施したものとする。図7（C）はこの変形した画像領域を表わすレイアウトプレーンの一部（はめ込み画像領域の開始位置）を示したものである。図中、“1”の部分はイメージのはめ込み領域を表わし、“0”の部分は、それ以外の領域であることを示している。このように、画像を変形した場合でも、レイアウトプレーンを作成することによって、従来では困難であった矩形画像以外でも画像ファイルは圧縮された状態のままにプリンタ側に送信できる。

【0042】但し、このような変形の場合、図8に示したように変形後の画像の横幅（widthとする）、縦幅（heightとする）が変化しないこと、および、画像の転送順序が変形後のラスタ順序と異なることが前提となる。

【0043】プリンタ側では、前述した説明通りにレイアウト情報に乗っ取って画像を復号し、レイアウトプレーンが“1”の部分に復号した画像情報をはめ込んでいけば良く、同じ構成で本効果が実現できる。

【0044】（第3の実施例）図9に本発明第3の実施例を示す。本実施例は前述した実施例では困難であった画像の回転にも対処したものである。

【0045】図10（A）は、今回作成する文書を示している。図中、破線で囲んだ部分がはめ込み画像を示し、前述の実施例同様に、本来は図10（B）の画像ファイルをアプリケーションソフト上で回転してはめ込みを行なうものである。

【0046】図10（C）はレイアウトプレーンの一部を示している。図中、“1”の部分が画像ファイルのは

め込まれる画素、“0”の部分はその以外の画素を示している。

【0047】この作成文書では、前述の実施例の項目で説明したように、はめ込み画像が回転している為に、格納している画像ファイルのラスタ方向と、はめ込んだ後のラスタ方向が異なっている。その為、図5の構成で実現しようとなると、プリンタ側に大量のバッファメモリを有して、復号後の画像情報を、バッファメモリ上のレイアウトプレーン対応画素位置に展開して格納するようになる。その為、プリンタの負荷は非常に大きくなる。

【0048】そこで、図9に示す構成を用いる。

【0049】図9において、圧縮画像2104は前述した第2実施例同様に、JPEG等の圧縮形式により圧縮されている画像を示している。復号化2051はこの圧縮画像を復号化する手段を示し、一度、ホストコンピュータ上で画像の復号を施す。

【0050】2052は符号化手段を示し、復号化した画像情報を再び、符号化する手段である。

【0051】この符号化手段2052はJPEGで用いているようなブロック状に符号化する方式ではなく、各画素ごとに符号化が行なえる、例えばDPCMのような符号化を施す。ここで、符号化する画素の順序であるが、格納されていた画像ファイルのラスタ方向の順序ではなく、回転された画像においてのラスタ方向の順序である。

【0052】図11を用いて説明する。図11(A)は図10(C)と同様で、図10(A)の作成文書にて作成されたレイアウトプレーンの一部である。図11

(B)は、このレイアウトプレーンの情報を受けて、符号化していく画素の順序を示している。

【0053】プリンタ側の画像復号化手段では、当然、符号化手段2052に対応した復号化手段であることは言うまでもない。

【0054】本実施例の構成では、回転のみならず、どのような変形を受けた場合においても実現できる。

【0055】また、本実施例は、圧縮画像2104は既に圧縮され、格納されている例として説明したが、これは圧縮画像でなくてもよいことは当然である。その場合には本実施例では復号器2051は不要になる。

【0056】(第4の実施例) 第4実施例では以下のような構成、即ち、複数のイメージ部のレイアウト情報を得るために、ホスト側でテキストデータにイメージ部のロケーションを指定するレイアウトプレーンを付加してから、プリンタに転送し、プリンタ側では復号化を行なう際にレイアウトプレーンの情報を利用してイメージの復号を制御する手段を設けることにより、プリンタにおける復号処理を容易なものとし、ハード構成も簡単化している。

【0057】図12は本実施例における画像出力装置を

示すブロック図である。図12は図5と同様ホストコンピュータとプリンタが接続されているシステムを簡単にブロック化したものである。図1と同じ働きをするブロックについては特に説明はしない。

【0058】PDL中に記述されているイメージのレイアウト情報はレイアウトプレーン生成部1103に入力される。レイアウトプレーンとは、図13に示す様に、テキスト領域は0、イメージ領域は1の値をもつ各画素1ビットのビットプレーンを指す。レイアウトプレーン生成部1103においては、入力されたレイアウト情報によってこのレイアウトプレーンを生成する。このときイメージは複数個存在した場合も、各イメージのレイアウト情報から1枚のビットプレーンにイメージ領域を図13のように設ける。

【0059】このようにして作られたレイアウトプレーンはさらに、符号化部1104に入力されてテキスト部と同様に算術符号化やランレングス符号化等の可逆符号化を用いて圧縮する。レイアウトプレーンの性質から上記のような可逆圧縮を行なっても、高い圧縮率で圧縮できるため、プリンタに転送する際のレイアウト情報自体の情報は非常に少ない。

【0060】さらに、レイアウトプレーンは符号化の際に、図14のようにテキストの色情報を示すnビットに、レイアウトプレーンの情報を1ビット付加して、n+1ビットとして符号化を行なう。さらに符号化されたあとにテキストとレイアウトプレーンは同じテキストメモリ1108に格納される。このレイアウトプレーンを付加する際に、図14のようにレイアウトプレーンのビットが0であったとき、すなわちその画素がテキストであったときには、テキスト部のnビットにはテキストの色情報が入る。また、レイアウトプレーンのビットが1であったとき、すなわちその画素がイメージであったときには、もしイメージとテキストの重なり合う場合を考えなければ、イメージの圧縮画像1101は色情報を含んでいるので、色情報は必要がない。そこでイメージを複数個張り付ける場合も考えて、テキスト部に複数のイメージを識別する情報を付加する。この識別情報を付加する処理はラスタライザ1102で行なう。イメージ識別情報とは、例えばホスト側からイメージを転送する転送順番を数値で持つ等の方法が考えられる。すなわちプリンタ側で複数のイメージを一意に識別できる情報をテキストの色情報の代わりに入れる。当然この時のイメージの識別方法は、あらかじめプリンタ側にその識別手段を設ける必要がある。

【0061】このようにしてホスト側で圧縮処理されたテキスト及びイメージの情報がプリンタの各メモリに格納される。ここでイメージは、従来例で述べたようにあらかじめ圧縮されてホストのハードディスク等に格納されてもよいし、圧縮されていないイメージをプリンタに転送する際にラスタライズと同時に圧縮処理を施しながら

ら転送してもよい。

【0062】テキストとイメージ、及びレイアウトプレーンはそれぞれ復号化部1111、1112で復号化される。このとき復号化部1111は常に復号処理を行なう。復号化部1111で復号化された $n+1$ ビットのテキスト情報およびレイアウトプレーンはイメージ識別回路1113に送られる。

【0063】イメージ識別回路1113を示したブロック図を図15に示す。入力されたテキストとレイアウトプレーンの情報の内、テキストの情報 n ビットについてはデコーダ802に入力される。また、同時にレイアウトプレーンの1ビットのデータは、信号線801によってデコーダ802に入力され、この信号がHighのとき、すなわちイメージの復号の際に、デコーダ802が働くように制御を行なう。デコーダ802は前記イメージ識別方法に従って、例えばホスト側から送られてくる順番がイメージ識別情報の値となっている場合には、その値によって指定された順番のイメージが、イメージメモリ内のどの箇所に格納されているかを示すメモリアドレスを出力する。今、ホストからA、B、Cという3つのイメージがこの順番にホストから転送され、図16のようにイメージメモリ内に格納されているとする。ここでAのイメージを復号する際に、イメージ識別情報には値“1”が入っている。この値に基づいて図15中のデコーダ802から、図12中のイメージメモリ1109の中のAのイメージが格納されているメモリ領域の先頭アドレスを出力する。他のB、Cのイメージを識別する際も同様に行なう。

【0064】このイメージ識別回路1113から出力されるメモリアドレス信号はメモリ制御部1110に入力される。このメモリ制御部1110では図16における各イメージの格納されているメモリ領域を指し示すポインタが、各イメージごとに管理されていて、メモリ制御部1110に入力されるアドレス情報から、どのイメージを読みだすかを判定し、その判定したイメージのポインタの指し示すメモリアドレスを出力する。このメモリアドレスがイメージメモリ1109に入力されて、該当するイメージのデータがメモリから読みだされて復号化部1112に入力される。

【0065】レイアウトプレーンはイメージ復号化の制御にも用いられる。すなわち、信号線1114よりレイアウトプレーンの情報1ビットがイメージの復号化部1112に入力される。テキスト部においてレイアウトプレーンのデータは“0”なので、イメージの復号化は行なわない。またイメージ部において、レイアウトプレーンのデータは“1”なので、イメージの復号化を行なう。またレイアウトプレーンはセクタ1115にも入力され、テキストとイメージの復号化されたデータを切り換えてプリンタエンジン1116に出力する。

【0066】本実施例により、イメージの復号の際に必

要となるレイアウト情報を簡単に得ることができ、また単純なハード構成で実現できる。

【0067】（第5の実施例）本発明の第5の実施例を示す図を図17に示す。図17は第4の実施例に示した図と同様、ホストコンピュータとプリンタが接続されているシステムを簡単にブロック化したものである。第4の実施例と同じ働きをするブロックについては特に説明しない。

【0068】複数のイメージを出力する場合、図6のように主走査方向の同じラインにイメージが複数ある際には第4の実施例にあるようなイメージを識別する処理を行なう必要があるが、図18にあるように主走査方向にイメージが重ならない時は、ホスト側から転送する際に図18の番号順に転送すれば、プリンタ側でイメージの復号処理を行なう際にも、イメージメモリに格納されている順に読みだせばよいので、イメージを識別する処理を行なう必要はない。ただし、画像出力の際、イメージの領域を指定する必要があるため、本実施例ではテキストの色情報の内、1色をイメージ領域を指定する情報として使用する。例えばテキストの色情報を表わすデータが n ビットの場合、 2^n 色の色を表わすことができるが、このうちの1色をイメージ領域を指定する情報に割り当てるため、色としては 2^{n-1} 色を表現できる。

【0069】図17のラスターライザ1002では、テキスト情報の内、テキスト部はビットイメージに展開し、イメージ部においては、テキストの色情報の内の決められた色（例えば白）を、イメージ領域を検出するために割り当てる。

【0070】第4の実施例と同様に、符号化処理されたテキスト情報と圧縮画像は、プリンタに転送されて、それぞれテキストメモリ1008、イメージメモリ1009に格納される。さらに画像出力時に各々復号化部1011、1012で復号化処理される。テキストの復号は常時行ない、その結果がイメージ検出部1013に入力される。このイメージ検出部では、ホスト側であらかじめイメージ領域検出のために割り当てられた色（イメージ領域色とする）を検出し、そのイメージ領域色が入力されたときには、イメージの復号化部1012に制御信号を送り、イメージの復号を行なう。また制御信号はセクタ1015にも入力され、テキスト情報とイメージ情報を切り換えてプリンタエンジン1016に出力する。

【0071】本実施例のような構成により、主走査方向にイメージが重ならない場合には、イメージを識別する必要がないので、テキストの色情報のうち、1色だけを割り当てることによりイメージ領域の指定ができる。本実施例ではレイアウトプレーンの役割をイメージ領域色で代用させていて、テキスト情報とは別にレイアウトプレーンを転送する必要がないので、その分メモリ容量が削減できる。

【0072】（第6の実施例）本発明の第6の実施例を示す図を図19に示す。図19は第4の実施例に示した図と同様、ホストコンピュータとプリンタが接続されているシステムを簡単にブロック化したものである。第2の実施例と同じ働きをするブロックについては特に説明しない。

【0073】複数のイメージを出力する場合、第4の実施例と同様に主走査方向にイメージが重ならない場合には、イメージ同士を区別する必要がない。そこで本実施例においてはレイアウトプレーンを用いることにより、テキスト情報のイメージ部の情報は転送を行なわない。従ってホストからプリンタに転送する全体の情報量を減らすことができる。すなわち図20に示す様に、テキスト情報の内斜線の付けられている領域については、データとしては何も転送をしない。

【0074】図19において、第4の実施例と同じように、PDL1200に記述されているレイアウト情報がレイアウトプレーン生成部1203に入力され、レイアウトプレーンが生成される。レイアウトプレーンは符号化部1205で符号化され、テキスト、イメージと共にプリンタに転送される。これらのデータは各々テキストメモリ1209、レイアウトメモリ1210、イメージメモリ1211に格納される。

【0075】復号化の際、レイアウトプレーンの復号を行なう復号化部1213は常に動作を行なう。その復号結果が、テキストとイメージの復号を行なう復号化部1212、1214にそれぞれ入力され、復号の制御を行なう。すなわち、テキスト部の復号を行なうときにはレイアウトプレーンのデータは“0”が出力されるので、否定回路1215により、復号化部1212に“High”の信号が入力されて、テキスト情報の復号を開始する。このときイメージの復号を行なう復号化部1214には“Low”の信号が入力されるので復号化は行なわない。イメージ情報の復号を行なう場合には、レイアウトプレーンのデータは“1”が出力されるので、イメージの復号化部1214には“High”の信号が入力され、逆にテキストの復号化部には“Low”の信号が入力されてイメージ情報のみの復号化が行なわれる。また、レイアウトプレーンはセレクト1216にも入力され、テキスト情報とイメージ情報を切り替えてプリンタエンジン1217に出力される。

【0076】本実施例の構成により、レイアウトプレーンを用いることによって、プリンタにおけるテキスト情報とイメージ情報の復号を制御することが可能になる。このためイメージ部のテキスト情報を転送する必要がなくなり、その分転送する情報量が減り、テキストを格納するメモリも削減できる。

【0077】（第7の実施例）本発明の第7の実施例を示す図を図21に示す。図21は第4の実施例に示した図と同様、ホストコンピュータとプリンタが接続され

ているシステムを簡単にブロック化したものである。第4の実施例と同じ働きをするブロックについては特に説明しない。

【0078】複数のイメージを出力する場合、第5、第6の実施例と同様に主走査方向にイメージが重ならない場合には、イメージ同士を区別する必要がない。従って本実施例においては、レイアウトプレーンを用いることにより、テキスト情報のイメージ部については、イメージの属性を変更するような情報をテキストの色情報を表すnビットの中に埋め込むような手段を設ける。

【0079】図21において、復号化の際には、テキストの復号化部1410は常時動作する。その出力はテキストデータnビットとレイアウトプレーンのデータ1ビットである。レイアウトプレーンはイメージの復号化部1411に入力されて復号の制御を行なう。復号化されたイメージデータはイメージ属性変更回路1412に入力され、同時に復号されたテキストデータも入力される。このイメージ属性変更回路1412に、例えばイメージの濃度を変更できるような回路構成を持たせることにより、入力されるテキストデータには濃度のレベル値となるような値をあらかじめホスト側で入れておくことによってイメージの濃度を変更することができる。このイメージ属性変更回路1412で変更されたイメージデータはセレクト1413に入力される。セレクト1413にはレイアウトプレーンのデータが制御信号として入力されて、テキストとイメージのデータを切り替えてプリンタエンジン1414に出力される。

【0080】また、イメージ属性変更回路の構成をいろいろ変えることにより、イメージのカラーバランスを変更したり、イメージにシェーディングをかける等の処理が可能になる。

【0081】本実施例の構成により、レイアウトプレーンを用いることによって、テキスト情報のイメージ部のデータに、イメージの属性を変更するような情報を埋め込んでデータの有効利用が図れる。またイメージのみの属性を変更するという従来不可能であった処理が本構成によって可能になる。

【0082】以上説明したように、本発明の上記第4～第7実施例によれば、ホストコンピュータでビットイメージに展開される文字・線画と、圧縮された自然画を別々にプリンタに転送し、フレームバッファメモリに格納してからプリンタエンジンに合わせて画像出力するような画像出力システムにおいて、複数の自然画を所定の位置に配置するための処理回路が簡単に構成できる。またその配置処理のために、プリンタに転送する情報に圧縮処理を行なうことにより転送処理時間も短縮でき、プリンタのメモリ容量も削減できる。

【0083】（第8の実施例）以下、本発明に係る画像処理装置の好適な第8の実施例を図22、23を用いて詳細に説明する。

【0084】図22において、一点鎖線で囲んだ部分はホストコンピューターを示し、破線で囲んだ部分は接続してあるプリンタを示す。この場合も説明を簡単にするためにホストコンピューターとプリンタは1対1で接続されているものとする。

【0085】図22中、3101はPDL（ページ記述言語）部を、3102はラスライザを、3103は符号化部を、3104～3106はホスト中に格納されている圧縮画像を示す。

【0086】しかしながら、ホストコンピューター上のアプリケーションソフトにおいて、図2に示すような1ページ分の文書を作成する場合、テキスト情報、即ち画像情報以外の文字や図形の領域情報はラスライズされ、符号化されて送信されるが、この時、画像領域を示す情報もPDL3101にて作成し符号化を行ない、送信する。この画像領域を示す情報とは、画像領域のスタート地点であるX座標のスタートアドレス、Y座標のスタートアドレス、画像の横幅サイズ、縦幅サイズ等のレイアウト情報を基に画像領域とそれ以外の領域を区別し、示したもので、例えば図6に示したように画像領域には1を画像領域以外には0を画素毎に置いた1ページ分の情報である。

【0087】以下、この情報をレイアウトプレーンとする。

【0088】また、プリンタにおいて、受信したテキスト情報はテキストメモリ3107に画像情報はイメージメモリ3108に、レイアウトプレーンはレイアウトプレーンメモリ3109に格納される。テキストメモリおよびイメージメモリからの復号化は、レイアウトプレーンメモリ内の情報を復号化しその情報を基に判断手段を用いて行なう。

【0089】即ち判断手段が復号化された図6のような情報を得た時、“0”の部分は画像領域以外と判断しスイッチ3114において端子Aを選択し、テキスト情報を復号化した後その情報をプリンタエンジン3115へと送る。

【0090】また、“1”である部分は画像領域と判断しスイッチ3114において端子Bを選択し、圧縮画像復号化部3113によって復号された画像情報をプリンタエンジン3115に送る。このようにしてラスター方向に順番に判断されプリンタエンジンへと送信される。

【0091】ところで、図7（a）のように1ページに複数の圧縮画像がはめ込まれている文書を作成する場合、上述によればレイアウトプレーンは図7（b）のようになる。これを使用しラスター方向へ順番に復号化を行なうと、Aの画像が全て送信されないうちにBの画像領域が現われ、どちらの画像領域かの判断はされないののでBの画像領域にメモリ内に残っているAの画像情報が送信されることとなる。

【0092】そこで、本発明では、1ページに複数の画

像領域がある場合、図23（c）～（e）に示すように1画像毎にレイアウトプレーンを作成する。

【0093】レイアウトプレーンはそれぞれ、上述同様符号化され、レイアウトプレーンメモリ3109に格納され、復号化されて、判断手段へと送られる。判断手段は、全てのレイアウトプレーン情報が0の場合、テキストメモリ3107の情報を復号化し、スイッチ3114において端子Aを選択し、プリンタエンジン3115へ送信する。

【0094】また、いずれかのレイアウトプレーンが1の場合、1であるレイアウトプレーンに対応する画像のイメージメモリ内を示す読み出しレジスタ（図24の3901または3902または3903）を選択し、選択されたレジスタを更新しつつ、そのレジスタが指す情報を圧縮画像復号化によって復号化した後、スイッチ3114にて端子Bを選択し、プリンタエンジン3115へと送信する。

【0095】このように画像毎にレイアウトプレーンをもつことによりどの画像の画像領域であるのか判断出来るので1ページに複数の画像領域があっても不具合を生じない。

【0096】（第9の実施例）次に、本発明の第9の実施例を図25、図26を用いて説明する。

【0097】本発明は、1ページに複数の圧縮画像がはめ込まれている場合でも、主走査方向に重なりを持たないものについては同一のレイアウトプレーンで画像領域を示すことを特徴とする。主走査方向に重なりを持たないとは、主走査方向には1つの画像領域しかない状態のことをいう。

【0098】即ち、図23（a）のように圧縮画像がはめ込まれた文書の場合、レイアウトプレーンは図25（a）、（b）になる。

【0099】図25はこのレイアウトプレーンの作成方法を示したフローチャートである。このフローチャートについて説明する。

【0100】まず、1ページの文書にはめ込まれている圧縮画像について、それぞれ画像領域の副走査方向のスタートアドレス、エンドアドレスの一覧表を作成する（s1）。

【0101】続いて、まず1つ目の画像の画像領域を第1プレーンに配置することに決める（s2）。（このときのプレーンとは、画像領域を示す前のレイアウトプレーンを指す）。

【0102】次にステップS5で、2つ目以降の画像は前述の一覧表を基に画像領域が主走査方向に重ならないと判断されれば、すなわち画像の副走査方向のスタートアドレス、エンドアドレスが他の画像のものと重ならない時には、同じ第1プレーンに（s6）、重なると判断された場合は第2プレーン・第3プレーン…と重ならないプレーンを調べてそれぞれの画像について、画像領域

をどのプレーン配置するか決める（s 4）。これを全面像の処理が終了するまで繰り返す（s 3, s 7）。

【0103】その後、それぞれの画像のレイアウト情報を決められたプレーンに展開することによって必要な枚数のレイアウトプレーンが作成される（s 8～s 11）。

【0104】図23（a）の文書を上述の方法にて作成された図26（a），（b）のレイアウトプレーンを用いて、図22の画像出力装置を動作させ、ラスタ方向に順番に判断しプリンタエンジンに送信した場合、Aの画像情報が全て送信される前にBの画像領域に入るが、この場合はレイアウトプレーンが分れているので前述の実施例に示した処理にて送信する画像情報を切り替える。次にCの画像領域が現われた時にはAの画像情報は全て送信されているので、切り替えをおこなわなくてもCが送信されるので同じレイアウトプレーンを用いてもなんら不都合を生じない。

【0105】以上説明したように本実施例によれば、画像領域と画像以外の領域を区別し示した情報を画像出力装置に転送することにより、復号化する際の処理構成を簡単にすることができる。

【0106】（第10の実施例）以下、本発明に係る画像処理装置の好適な実施例を図27～30を用いて詳細に説明する。

【0107】図27において、一点鎖線で囲んだ部分はホストコンピュータを示し、波線で囲んだ部分は接続してあるプリンタを示す。この場合も説明を簡単にするためにホストコンピュータとプリンタは1対1で接続されているものとする。

【0108】図27中、5101はPDL（ページ記述言語）部を、5102はラスタライザを、5103は符号化部を、5104はホスト中に格納されている圧縮画像を示す。5101～5104についての詳細は従来例と同様である。

【0109】しかしながら、図2に示すような1ページ分の文書を作成する場合、画像情報以外のテキスト情報や図形等のグラフィック情報は従来と同様にラスタライズされ、符号化されて送信されるが、この時、画像領域を示すレイアウトプレーン情報並びに、SOIプレーンやPDL5101にて作成されたテキスト情報と共に、符号化部5103にて符号化が行なわれ、セクタ5105及びケーブル5106を経由してプリンタに送信される。一方、画像情報は圧縮されたままの状態、プリンタに出力される。

【0110】プリンタにおいて、受信したテキスト情報はテキストメモリ5108に画像情報はイメージメモリ5109に格納される。

【0111】テキストメモリに格納された、テキスト情報、レイアウトプレーン情報並びにSOIプレーン情報は、テキスト復号化部5111にて同時に復号される。

復号されたテキスト情報、レイアウトプレーン情報、SOIプレーン情報はそれぞれ信号線5120、5121、5122に出力される。

【0112】同時に復号された、前記3つの情報の内、レイアウトプレーンの情報に基づき、画像情報の配置が決まる。

【0113】例えば、レイアウトプレーンの情報が図3の場合には、破線で囲まれた“1”の領域に画像が配置される。

【0114】この位置に配置される画像情報は、イメージメモリ5109に格納された圧縮画像情報を画像復号化部5112にて復号することにより得られる。該画像復号化部の動作は、該レイアウトプレーンの情報によって制御される。即ち、レイアウトプレーン情報の値が“1”の時に、復号化動作を行ない、“0”の時には、復号化動作を停止する。

【0115】また、該レイアウトプレーンの情報によりセクタ5113も制御され、復号化動作が行なわれている時には、復号画像情報がエンジンに出力されるようになっている。逆に、復号化動作が行なわれていない時には、テキスト復号化部5111の出力がプリンタエンジン（114）に出力される。

【0116】復号する画像が複数ある場合には、前記制御以外に、何番目の画像を復号するのかといった制御も必要になる。その制御を行なうのが画像識別信号生成部5115とメモリ管理部5110である。

【0117】メモリ管理部5110は、何番目の圧縮画像情報が、イメージメモリ5109のどの位置（番地）に格納されているかを管理しているところで、画像識別信号生成部より送られてくる画像番号に対応した圧縮画像情報をイメージメモリ5109から読み出し、該画像情報を画像復号化部5112に送る。

【0118】画像識別信号生成部5115は、画像復号化部5112において復号する画像の番号を識別するところである。これまで説明したレイアウトプレーン情報だけでは、画像の配置場所は分かっても、画像の番号までは分からない。

【0119】そこで、SOIプレーンを利用する。SOIプレーンは、図28に示すように、画像の開始位置（左上の位置）を“1”、その他の領域を“0”で表わしたビットマップ情報である。また、画像を配置しない領域の値はもちろん“0”である。

【0120】“1”の個数は、配置する画像の数に等しく、1ページ内のN個の画像を配置する場合、“1”の個数はN個となる。

【0121】このSOIプレーン上の“1”の情報に基づき、配置する画像の番号を画像識別信号生成部5115で生成する。

【0122】該画像識別信号生成部5115のブロック図を図29に示す。

【0123】同図において、1501はSOIプレーン情報を入力する端子、1502はレイアウトプレーン情報を入力する端子、1503はSOIプレーン内の“1”を検出して、その数をカウントするmビットのカウント、1504は主走査方向の画素分の容量を持つmビット幅のFIFOメモリ、1505は前記FIFOメモリのmビット出力をゼロにマスクするマスク回路、1506は前記マスク回路出力とカウンタ出力を切り換えるセクタ、1507は画像番号を出力する端子、1508はNOT素子、1509、1510は2入力NAND素子でR-Sフリップフロップを構成する。

【0124】また、1511は該R-Sフリップフロップの出力信号である。

【0125】ここに、図30(a)に示すレイアウトプレーン情報と図30(b)に示すSOIプレーン情報が入力されたとする。入力される順序はページの左上からラスタースキャン走査される。

【0126】まず最初に、不図示の回路によりカウンタ1503がゼロにクリアされると共に、R-Sフリップフロップの出力信号1511もLowレベルに設定される。

【0127】次に、両プレーン共、最初の画像配置位置に到達するまで、“0”が入力され続ける。この間、画像番号出力端子1507には、マスク回路1505でゼロにマスクされたmビットの信号が出力され続ける。

【0128】やがて、最初の画像Aの配置位置に到達するとSOIプレーン情報が“1”になり、カウンタ1503がカウントアップして“1”という値になる。また、R-Sフリップフロップがセットされ出力信号1511がHighレベルに設定される。

【0129】これにより、カウンタ1503の出力信号値“1”が、画像番号としてセクタ1506で選択され、端子1507に出力される。

【0130】また、セクタ1506の出力はFIFOメモリ1504にも入力される。

【0131】SOIプレーン情報が“1”になった直後は、レイアウトプレーン情報の値もしばらく“1”が続くが、画像の横方向のサイズを越したところで“0”に戻る。ここで、R-Sフリップフロップがリセットされ出力信号1511がLowレベルになり、マスク回路1505でゼロにマスクされたmビットの信号が、また出力される。

【0132】さらに、走査が進んで行くと、次の画像Bの配置位置に到達し、またSOIプレーン情報が“1”になる。今度はカウンタ1503の値が“2”になり、その値が画像番号として、セクタ1506を経由し端子1507に出力される。一方、該出力はFIFOメモリ1504にも入力される。

【0133】画像Aと同様、画像の横サイズを越すまで前記状態が続き、サイズを越したところで端子1507

の出力はゼロに戻る。やがて該ラインの走査が終了し、次のラインの走査が始まる。

【0134】次のラインでは、SOIプレーン情報に“1”が1つも無いので、レイアウトプレーン情報のみで、該画像識別信号生成1115は制御される。1ライン前の画像番号は、FIFOメモリ1504に記憶されていて、正確に1ライン遅延したタイミングで、該FIFOから読みだされてくる。

【0135】該画像番号情報は、マスク回路1505に入力されるが、画像配置領域ではレイアウトプレーン情報は“1”であるため、該マスク回路を素通りし、セクタ1506を経由し、前ラインと同じ画像番号情報が、端子1507から出力される。また、該出力情報は、FIFOメモリ1504に再入力される。

【0136】画像A、画像B共に、画像配置領域が垂直方向（下方向）に続く限り、上記処理が繰り返し行なわれる。

【0137】そして、画像Aの配置領域が終了した次のラインの走査を行なうものとする。この時、FIFOメモリ1504には、まだ画像Aに対応した画像番号が記憶されている。しかし、端子1502から入力されるレイアウトプレーン情報には、画像Aの配置を示す情報が無い、即ち、端子1502から入力される信号は“0”のままである。

【0138】よって、FIFOメモリ1504から1ライン遅延されて読みだされたmビットの信号はマスク回路1505によりゼロにマスクされ、セクタ1506を経由し、端子1507に出力される。一方、画像Bの配置領域は継続しているため、走査が該領域に達すると端子1502から入力されるレイアウトプレーン情報は“1”になり、FIFOメモリ1504に記憶されている、画像Bに対応した画像番号がマスク回路1505を素通りし、該画像番号はセクタ1506を経由し、端子1507から出力される。

【0139】以後、画像Bの配置領域に関しては、該領域が垂直方向に継続する限り、上記処理が繰り返し行なわれる。そして、画像Bの配置領域が終了した次のラインの走査において、前記、画像Aの処理のごとくFIFOメモリ1504から出力される画像番号はマスク回路1505にて、ゼロにマスクされ端子1507から出力される。

【0140】該ラインの走査が終了した時点では、FIFOメモリ1504の内容は全てゼロにクリアされる。この状態は、次の画像配置領域、即ちSOIプレーン情報が“1”の領域に達するまで続く。

【0141】ライン走査が進んで行き、やがて画像Cの配置領域に到達すると、SOIプレーン情報が“1”になり、カウンタ1503がカウントアップして“3”という値になる。また、R-Sフリップフロップがセットされ、出力信号1511がHighレベルに設定され

る。

【0142】後は、既に画像A、画像Bのところで説明した内容とまったく同じ動作で、画像Cの配置領域に対応したエリアのみにおいて、画像Cの画像番号が端子1507から出力される。

【0143】結局、図30(a)のレイアウトプレーン情報、図30(b)のSOIプレーン情報に基づいて、図30(c)に示す画像番号情報が、画像識別信号生成部1115にて生成され、メモリ管理部1110に送られて、画像の配置が確実に行なわれる。

【0144】(第11の実施例)次に、本発明の第11の実施例を図31を用いて詳細に説明する。

【0145】本実施例は、図27における画像識別信号生成部5115の他の構成方法について述べるものである。よって、図27のシステム構成と同様であり、その動作説明は省略する。

【0146】図31において、図29中のブロックと同じ機能を有するブロック、あるいは、同じ機能を有する入出力端子には、同一番号を付し説明を省略する。図31で図29と異なるブロックは、ゼロ検出部1901と2入力AND素子1902である。

【0147】ゼロ検出部1901は、FiFoメモリ1504から出力されるmビットの信号が全て“0”であることを検出するものであり、これにより、1ライン前が画像の配置領域でなかったことが分かる。

【0148】該検出結果を2入力AND素子1902でもって、端子502から入力されるレイアウトプレーン情報と論理積をとることにより、画像の配置領域が、現在走査中のラインから新たに始まることが分かる。論理積の結果は信号線1903とする。

【0149】本実施例では、まず最初に、カウンタ1503のクリアとFiFoメモリ1504のゼロクリアが必要である。

【0150】前記実施例と同様、図30(a)に示すレイアウトプレーン情報と、図30(b)に示すSOIプレーン情報が入力されるものとする。

【0151】まず、両プレーン共、最初の画像配置位置に到達するまで、“0”が入力され続ける。この間、信号線1903はLowレベルなので、画像番号出力端子1507には、マスク回路1505のmビットの出力信号(もちろん、この値はゼロである。)が出力される。

【0152】やがて、最初の画像Aの配置位置に到達するとSOIプレーン情報が“1”になり、カウンタ1503がカウントアップして“1”という値になる。

【0153】この時、レイアウトプレーン情報も“1”になっている。また、FiFoメモリ1504のmビット出力はゼロなので、それが、ゼロ検出部1901で検出され、AND素子1902の出力信号1903がHighレベルになる。

【0154】これにより、カウンタ1503の出力信号

値“1”が、画像番号としてセクタ1506で選択され、端子1507に出力される。また、セクタ1506の出力はFiFoメモリ1504にも入力される。

【0155】SOIプレーン情報が“1”になった直後は、レイアウトプレーン情報の値もしばらく“1”が続くが、画像の横方向のサイズを越したところで“0”に戻る。ここで、AND素子1902の出力信号1903もLowレベルとなり、マスク回路1505でゼロにマスクされたmビットの信号が、また出力される。

【0156】さらに、走査が進んで行くと、次の画像Bの配置位置に到達し、またSOIプレーン情報が“1”になる。今度はカウンタ1503の値が“2”になり、その値が画像番号として、セクタ1506を経由し端子1507に出力される。一方、該出力はFiFoメモリ1504にも入力される。

【0157】画像Aと同様、画像の横サイズを越すまで前記状態が続き、サイズを越したところで端子1507の出力はゼロに戻る。やがて該ラインの走査が終了し、次のラインの走査が始まる。1ライン前の画像番号は、FiFoメモリ1504に記憶されていて、正確に1ライン遅延したタイミングで、該FiFoから読みだされてくる。読み出されてくる内容は、画像配置領域では、各画像に対応したゼロ以外の画像番号であり、画像配置領域以外では、ゼロという値である。

【0158】よって、ゼロ検出部1901におけるゼロの検出タイミングとレイアウトプレーン情報が“1”になるタイミングは、まったく逆の関係になるため、AND素子出力信号1903はいつもLowレベルとなり、マスク回路1505の出力がセクタ1506を経由して、画像番号出力端子1507に出力される。

【0159】画像A、画像B共に、画像配置領域が垂直方向(下方向)に続く限り、上記処理が繰り返行なわれる。

【0160】そして、画像Aの配置領域が終了した次のラインの走査を行なうものとする。この時、FiFoメモリ1504には、まだ画像Aに対応した画像番号が記憶されている。しかし、端子1502から入力されるレイアウトプレーン情報には、画像Aの配置を示す情報が無い、即ち、端子1502から入力される信号は“0”のみである。

【0161】よって、FiFoメモリ1504から1ライン遅延されて読みだされたmビットの信号はマスク回路1505によりゼロにマスクされ、セクタ1506を経由し、端子1507に出力される。

【0162】一方、画像Bの配置領域は継続しているため、走査が該領域に達すると端子502から入力されるレイアウトプレーン情報は“1”になり、FiFoメモリ1504に記憶されている、画像Bに対応した画像番号がマスク回路1505を素通りし、該画像番号はセクタ1506を経由し、端子1507から出力される。

【0163】以後、画像Bの配置領域に関しては、該領域が垂直方向に継続する限り、上記処理が繰り返行なわれる。そして、画像Bの配置領域が終了した次のラインの走査において、前記画像Aの処理のごとく、FiFoメモリ1504から出力される画像番号はマスク回路1505にて、ゼロにマスクされ端子1507から出力される。

【0164】該ラインの走査が終了した時点ではFiFoメモリ1504の内容は全てゼロにクリアされる。この状態は、次の画像配置領域、即ちSOIプレーン情報が“1”の領域に達するまで続く。

【0165】ライン走査が進んで行き、やがて画像Cの配置領域に到達すると、SOIプレーン情報が“1”になり、カウンタ1503がカウントアップして“3”という値になる。また、ゼロ検出部1901において、FiFoメモリ1504から出力されるゼロが検出され、レイアウトプレーン情報は“1”が入力されるため、番号1903がHighレベルに設定される。

【0166】後は、既に画像A、画像Bのところで説明した内容とまったく同じ動作で、画像Cの配置領域に対応したエリアのみにおいて、画像Cの画像番号が端子1507から出力される。

【0167】結局、図30(a)のレイアウトプレーン情報、図30(b)のSOIプレーン情報に基づいて、図30(c)に示す画像番号情報が、図31の画像識別信号生成部にて生成され、図27のメモリ管理部5110に送られて、画像の配置が確実に行なわれる。

【0168】以上説明したように、本実施例によれば、画像の配置領域を示すレイアウトプレーン情報と画像のスタート位置を示すSOIプレーン情報をPDL展開時に生成し、PDL展開して出来たテキスト情報と前記情報とを同一の符号化回路で符号化し、それをホストコンピュータからプリンタへ転送し、プリンタ側では、逆の復号化を行なう。

【0169】そして、復号化して得られた前記レイアウトプレーン情報とSOIプレーン情報から、画像識別信号部において、画像識別信号を生成することにより、複数画像のレイアウトを容易に管理できるようになった。

【0170】その結果、従来よりも安価なコストで、柔軟な機能を実現出来るようになった。

【0171】(第12の実施例) 上述の従来例で述べたプリントシステムのうち、プリンタ記述言語(PDL)プログラム、バイトマップイメージをホストコンピュータで生成し、その情報をプリンタに送り、プリンタ側のコントローラでこれらを展開、配列し、ハードコピーを作るものについては、下記のような欠点があった。

(1) バイトマップイメージをホストからプリンタに送る時、その量が大きいため時間がかかる。または広帯域伝送路が必要となる。

(2) コードを展開するとき、ホスト側とプリンタ側に

フォントをそれぞれ必要とし、構成部の共用化ができない。

(3) プリンタ側のプログラム展開部はファームで作られており、バージョンアップができない。

(4) プリンタ側のプログラム展開部はファームで構成されているため、拡張性がない。

(5) プリンタの稼働率は一般にホストに比べ低いため、プリンタ側のハード資源の有効利用ができない。

【0172】かかる点に鑑みて、本実施例は図32に示す構成をとる。アプリケーションプログラム4001により文書を生成し、文書を生成するとき必要なファイルを通信ライン4002、通信マネージャ4003を介し送受する。

【0173】生成された文書はOS4004、イメージマネージャ4005、CRTドライバ4006、モニタTV4007を介しモニタ表示することができる。またデバイスドライバ4008を介してプリンタ部に文書を送りエンジンコントローラ4009、エンジンI/F4010、プリンタエンジン4011を用いてハードコピーを得ることもできる。

【0174】イメージマネージャ4005の詳細なブロック図を図33に示す。イメージマネージャ4005はアプリケーションで生成した文書ファイルをOS4004を介し、受け取る。文書ファイルはPDLファイルとイメージ圧縮ファイルから構成されている。PDLファイルは、インタプリンタ4012、ラスライザ4013でフォント部4014に記憶されたフォントを用いて展開メモリ4015に展開される。

【0175】イメージ情報は圧縮されたファイルで受け取り、イメージ伸長部4016で伸長される。PDLファイルには、このイメージファイルの編集コマンド、属性が含まれており、インタプリンタ4012でそれらを分離し属性情報はイメージ属性メモリ4017に記録される。イメージは編集処理部4018とワークメモリ4019を用い、編集コマンドに従って1次変換が行われ展開メモリ4015に書き込まれる。展開メモリ4015の内容は、CRTドライバ4006を介しモニタTV4007に送られ、その内容が逐次モニタされる。

【0176】またレイアウト制御信号はレイアウトメモリ4020に記憶される。これら処理により、最終的な文書が展開メモリ4015に生成される。イメージマネージャ4005から文字線画、高能率に圧縮されたイメージ、レイアウトデータ、イメージ属性情報が得られる。

【0177】デバイスドライバの機能を図34に示したイメージマネージャ4005から送られた文字/線画、レイアウトデータ、イメージ属性情報をマルチプレックス4021でマルチプレックスし、ロスレス圧縮部4022でロスレス(可逆)の圧縮を行う。さらにこの信号とイメージ(圧縮されたもの)をマルチプレックス4023でマルチプレックスし通信部4024を介してプリン

タ部に送出する。

【0178】プリンタ部のエンジンコントローラ4009の機能を図38に示した。ホストコンピューター部から送られてきた信号を通信部4025で受け、デマルチプレクサ4026で分離し、文字／線画、レイアウトデータ、イメージ属性情報はメモリ1に、イメージ（圧縮）情報はメモリ2に記録される。

【0179】デコーダ1で文字／線画情報を伸長し、デコーダ2でイメージ（圧縮）情報を伸長する。レイアウトデータ、イメージ属性情報もデコーダ1で解読され、レイアウト情報で合成部4027の制御を、イメージ属性情報で解像度変換部の制御を行う。前者は文字／線画とイメージ情報4028の合成を行い、後者はイメージファイルのドット密度をエンジンの解像度に変換する。これらの処理はタイミング信号発生部4029によりエンジン動作に同期して、リアルタイムに行われる。

【0180】（第13の実施例）図36にデバイスドライバの他の実施例を示す。文字／線画が2値の場合、情報量は小さく、またレイアウト、イメージ属性情報もその量は小さく、圧縮しない方がコストパフォーマンスが上がる場合がある。この場合は、文字／線画、レイアウト、イメージ属性、イメージ（圧縮）情報を直接マルチプレクサ4030によりマルチプレックスし、通信部4031を介してプリンタ部へ送る。

【0181】プリンタ部エンジンコントローラ部のデコーダ1において文字／線画に対する伸長機能は不要となりハード的な縮小が図れる。

【0182】以上説明したように、第12、第13実施例によれば、ホストコンピューターとプリンタ間の情報は圧縮されたデータで送られるため、伝送時間が短縮される。またビットイメージへの展開はホスト側で行われ、その時に必要なフォントはホスト側にのみあればよく、従来プリンタ側に必要であったものが不要となり資源の有効利用が可能となる。さらにプリンタ側にあったフレームメモリも圧縮比になり、コストの低減が図れる。コンピュータでの処理もマルチタスクで行い、ラストライズ処理も1つのタスクとして処理すれば、ホストコンピューターの効率向上が図れる。

【0183】

【発明の効果】以上の様に、本発明によれば、画像のはめ込みを行う場合に効率の良い画像の転送及び再生が可能となる。

【図面の簡単な説明】

【図1】本発明の画像出力システムの第1の実施例の構成を示すブロック図。

【図2】作成する文書の一例を示す図。

【図3】テキスト情報とイメージ情報の復号処理を行う領域を説明する図。

【図4】図2の構成におけるイメージ領域指定部の構成を示すブロック図。

【図5】本発明の第2の実施例の構成を示すブロック図。

【図6】レイアウトプレーンを説明する図。

【図7】はめ込み画像を変形させた場合の説明図。

【図8】図17における変形条件の説明図。

【図9】本発明の第3の実施例の構成を示すブロック図。

【図10】はめ込み画像を回転させた場合の説明図。

【図11】符号化画素の順序を説明した図。

【図12】本発明の第4の実施例の構成を示すブロック図。

【図13】レイアウトプレーンを説明する図。

【図14】レイアウトプレーンとテキスト情報との関係を示す図。

【図15】第1の実施例におけるイメージ識別回路の構成を示すブロック図。

【図16】複数のイメージ情報がイメージメモリに格納されている様子を示す図。

【図17】本発明の画像出力システムの第5の実施例の構成を示すブロック図。

【図18】第5の実施例における複数のイメージの転送順序を示す図。

【図19】本発明の画像出力システムの第6の実施例の構成を示すブロック図。

【図20】第6の実施例における全転送情報を示す図。

【図21】本発明の画像出力システムの第7の実施例の構成を示すブロック図。

【図22】第8の実施例の構成を示すブロック図。

【図23】第8の実施例におけるレイアウトプレーンの説明図。

【図24】イメージメモリの概略を示す図。

【図25】第9の実施例におけるレイアウトプレーンを作成するためのフローチャートを示す図。

【図26】第9の実施例におけるレイアウトプレーンを示す図。

【図27】本発明の第10の実施例の構成を示す図。

【図28】前記文書のSOIプレーン情報を示す図。

【図29】画像識別信号生成部5115を示す図。

【図30】3つの画像を含む文書のレイアウトプレーン情報、SOIプレーン情報、及び各画像に付される画像番号を示す図。

【図31】本発明の第11の実施例である画像識別信号生成部のブロック図。

【図32】本発明の第12の実施例を示す図。

【図33】イメージマネージャの機能を示す図。

【図34】デバイスドライバの機能を示す図。

【図35】エンジンコントローラの機能を示す図。

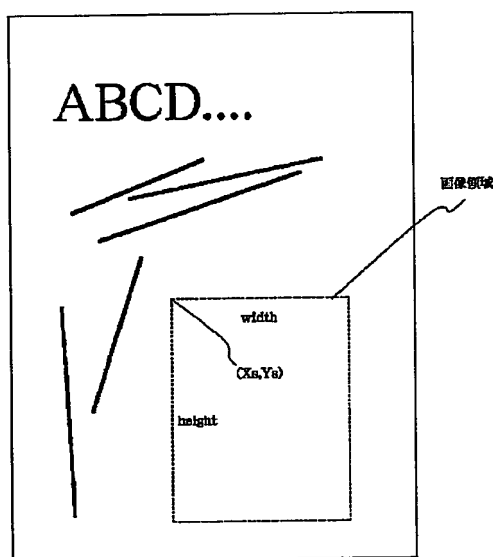
【図36】デバイスドライバの第13の実施例を示す図。

【符号の説明】

200 PDL展開部 (PDL: ページ・記述・言語)
 201 圧縮された画像情報の格納部
 202 ラスタライザ
 204 符号化部
 205 セレクタ
 206 ホストコンピュータとプリンタを接続するケーブル
 208 符号化されたテキスト関係の情報を格納するメモリ
 209 ホストから受け取った画像情報を格納するイメージメモリ

210 該イメージメモリを管理するブロック
 211 テキスト関係の情報の復号化部
 212 圧縮された画像情報の復号化部
 213 テキスト情報と画像情報を切り換えるセレクタ
 214 プリンタエンジン
 503 SOIプレーン中の“1”をカウントするカウンタ
 504 画像番号情報を1走査分蓄えるFIFOメモリ
 505 マスク回路
 901 ゼロ検出部

【図2】



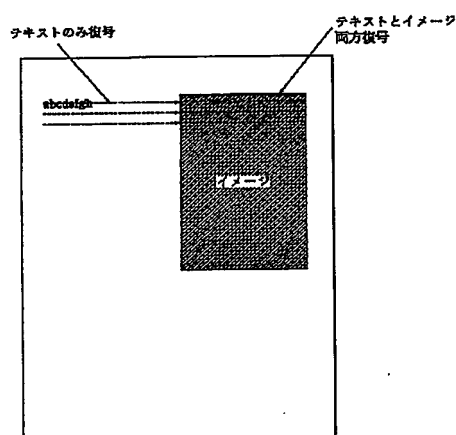
【図11】

.....
00000000000000	00000000000000
00000000000000	00000000000000
00011000000000	00012000000000
00111111100000	003456780000
00111111111110	009
01111111111111	
01111111111111	
.....	

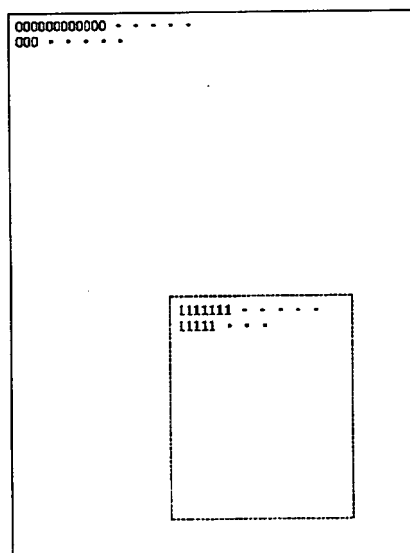
(A) レイアウトプレーンの一部

(B) 符号化要素の順序

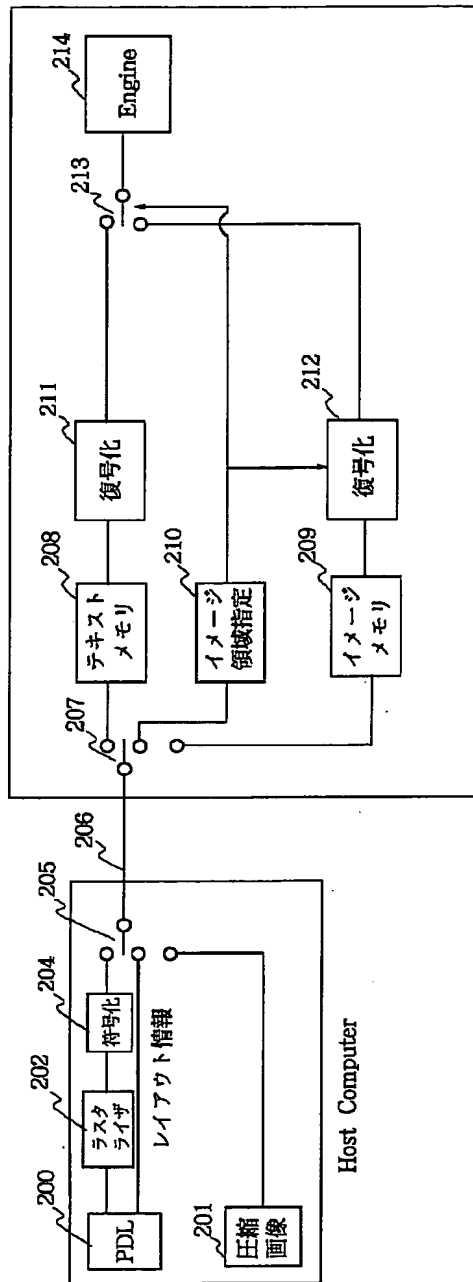
【図3】



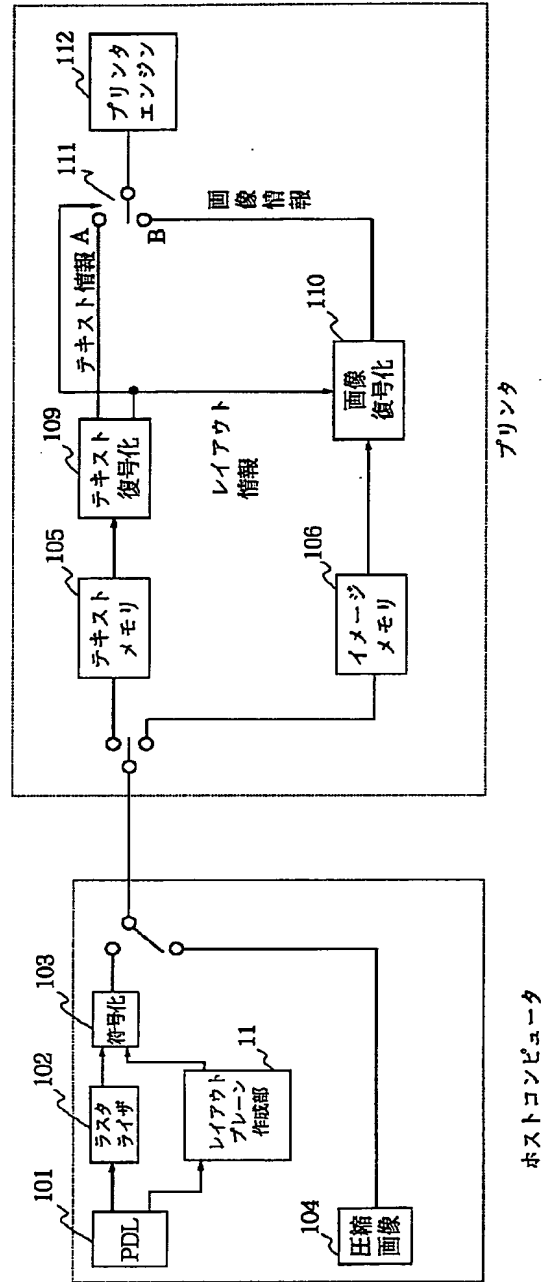
【図6】



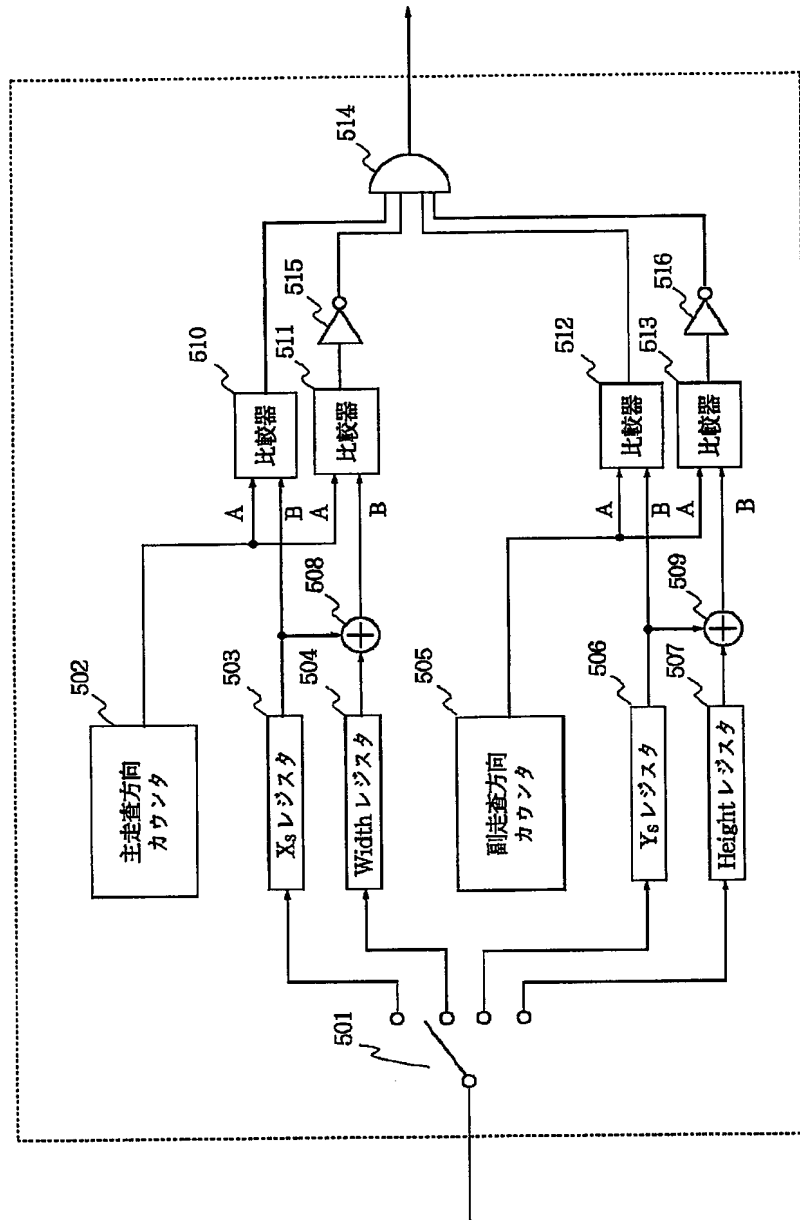
【図 1】



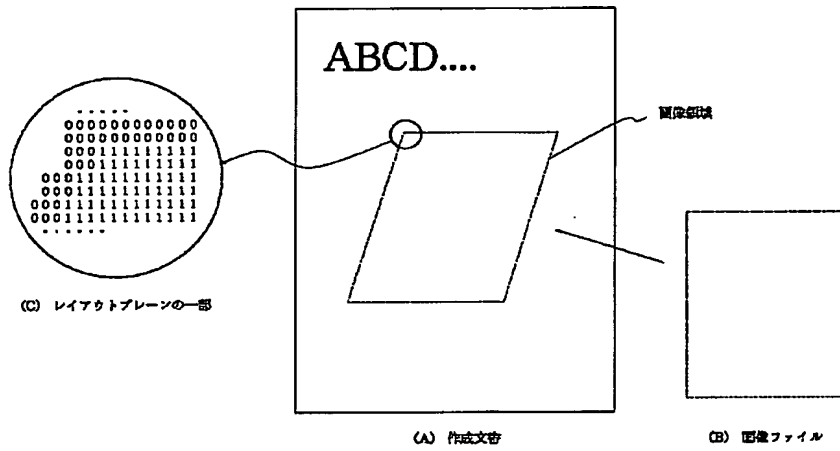
【図 5】



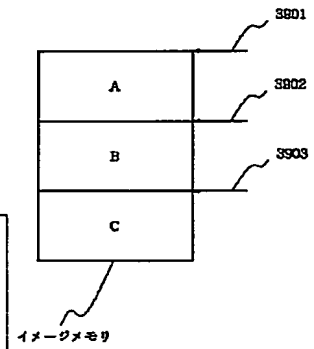
【図4】



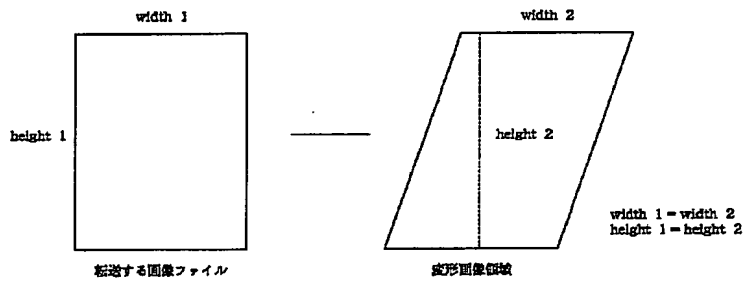
【図 7】



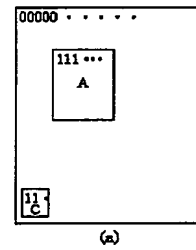
【図 24】



【図 8】



【図 26】



【図 10】

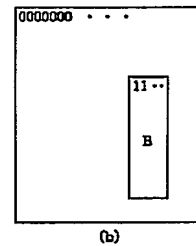
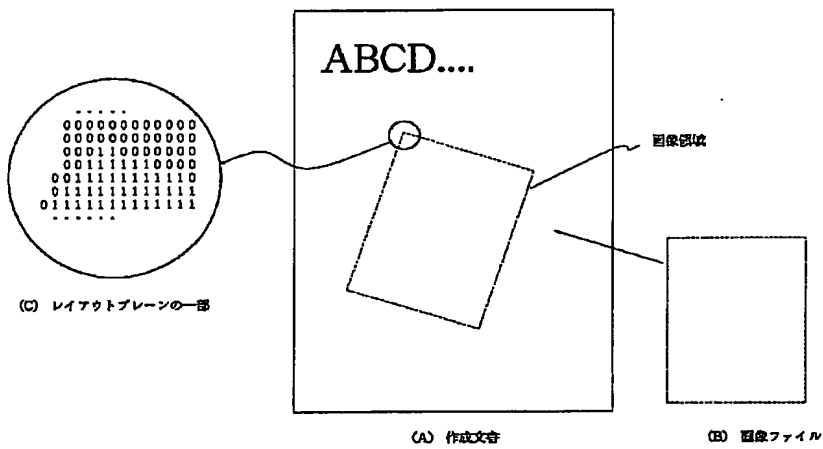


Figure 1 is a block diagram of a printing system. The system is divided into two main sections. The left section (2101) contains a PDL (2101), a rasterizer (2102), a compression unit (2104), a decompression unit (2105), a layout creation unit (2106), and a compression unit (2107). The right section (2108) contains a text encoding unit (2109), a text decoding unit (2110), an image encoding unit (2111), and a printer engine (2112). Data flows from PDL to rasterizer, then to compression unit (2104), then to decompression unit (2105), then to layout creation unit (2106), then to text encoding unit (2109), then to text decoding unit (2110), then to image encoding unit (2111), and finally to printer engine (2112). There are also direct connections from rasterizer to text encoding unit (2109) and from layout creation unit (2106) to image encoding unit (2111).

ホストコンピュータ

The diagram illustrates a system architecture for a printer, divided into two main sections: a Host Computer and a Printer.

Host Computer Section:

- 1100:** A bus line connecting the Host Computer to the Printer.
- 1101:** A PDL (Page Description Language) block.
- 1102:** A rasterizer block (ラスタライザ).
- 1103:** A character generator block (文字生成部).
- 1104:** An image processor block (レライアウトプレーン生成部).
- 1105:** A character buffer block (符号化).
- 1106:** An image buffer block (イメージ).

Printer Section:

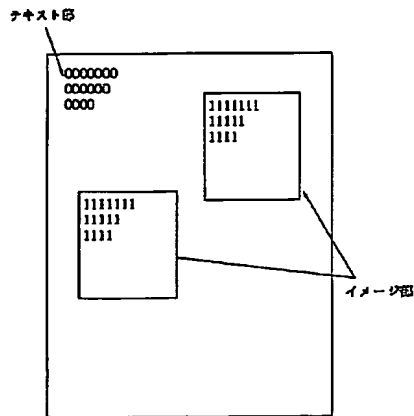
- 1107:** A switch or connector point.
- 1108:** A text memory block (テキストメモリ).
- 1109:** An image memory block (イメージメモリ).
- 1110:** A control block (制御).
- 1111:** A character buffer block (符号化).
- 1112:** An image buffer block (イメージ).
- 1113:** A character buffer block (符号化).
- 1114:** A switch or connector point.
- 1115:** An image buffer block (イメージ).
- 1116:** An engine block (エンジン).

Data Flow and Connections:

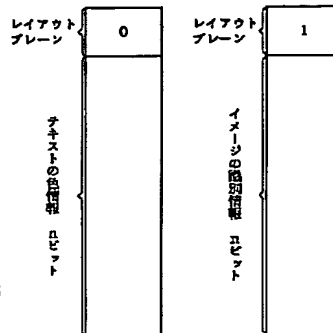
- The Host Computer's PDL (1101) sends data to the rasterizer (1102).
- The rasterizer (1102) sends data to the character generator (1103).
- The character generator (1103) sends data to the image processor (1104).
- The image processor (1104) sends data to the character buffer (1105).
- The character buffer (1105) sends data to the switch (1106).
- The switch (1106) sends data to the printer's text memory (1108).
- The printer's text memory (1108) sends data to the character buffer (1111).
- The character buffer (1111) sends data to the switch (1114).
- The switch (1114) sends data to the printer's image memory (1113).
- The printer's image memory (1113) sends data to the image buffer (1112).
- The image buffer (1112) sends data to the printer's image memory (1109).
- The printer's image memory (1109) sends data to the control block (1110).
- The control block (1110) sends data to the image buffer (1111).
- The control block (1110) sends data to the engine (1116).

Host Computer

【図13】

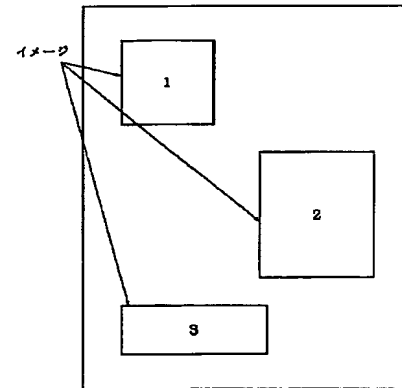
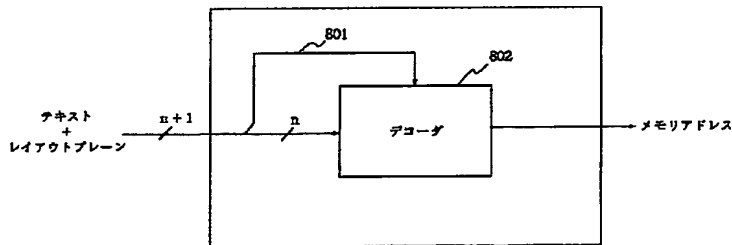


【図14】

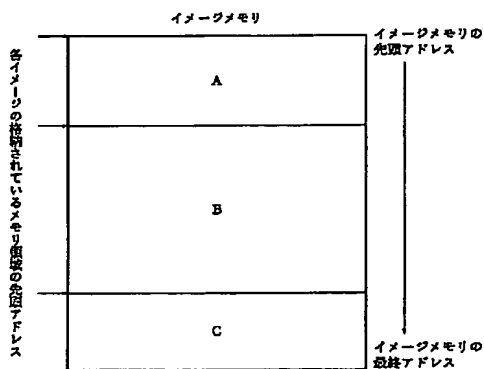


【図18】

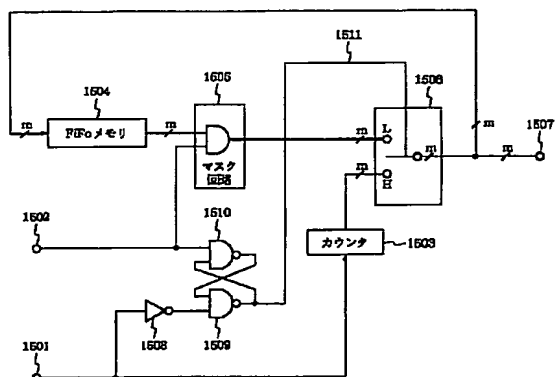
【図15】



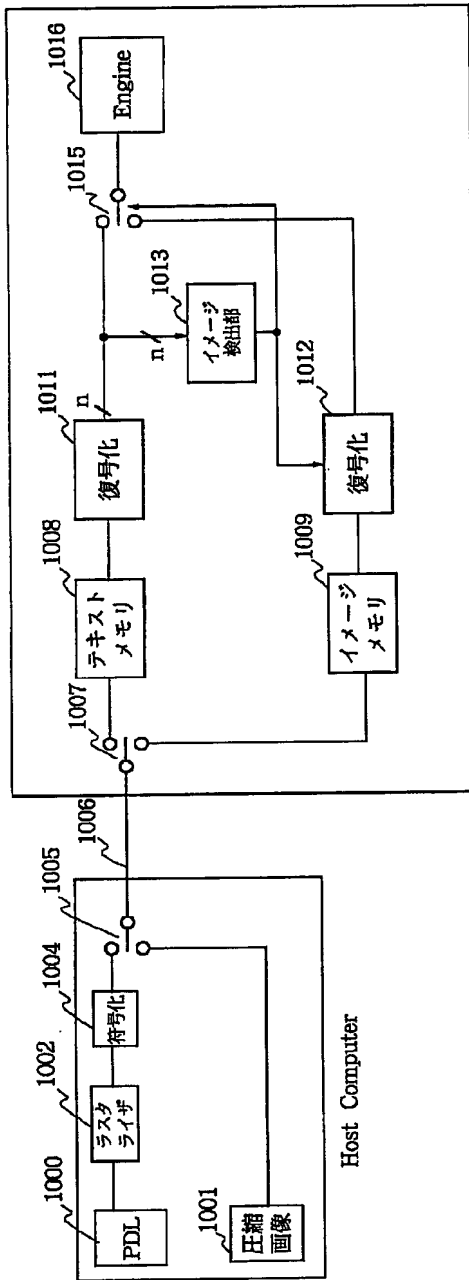
【図16】



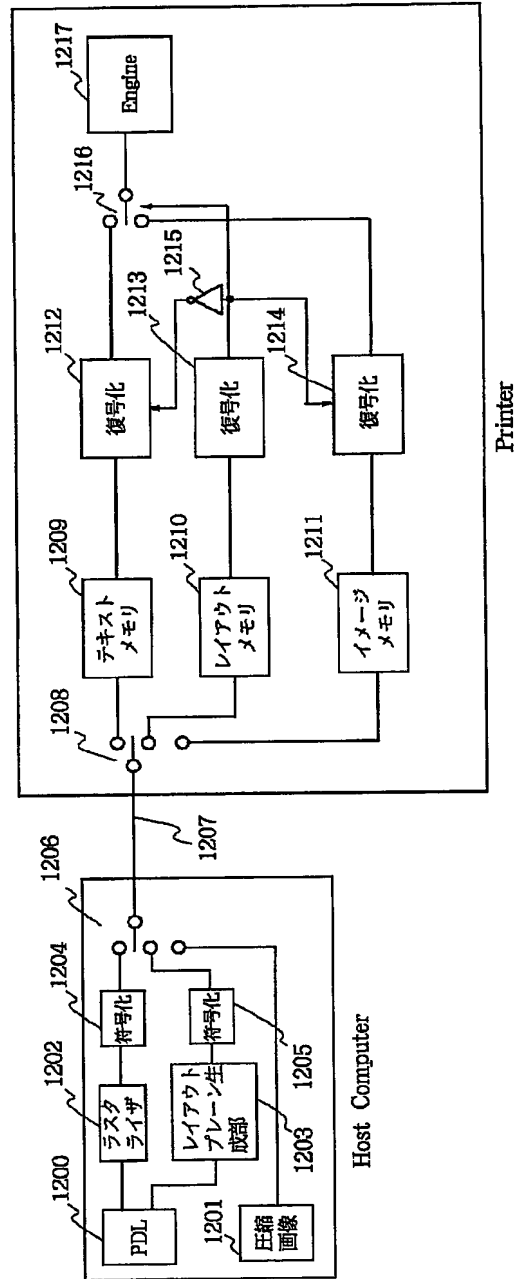
【図29】



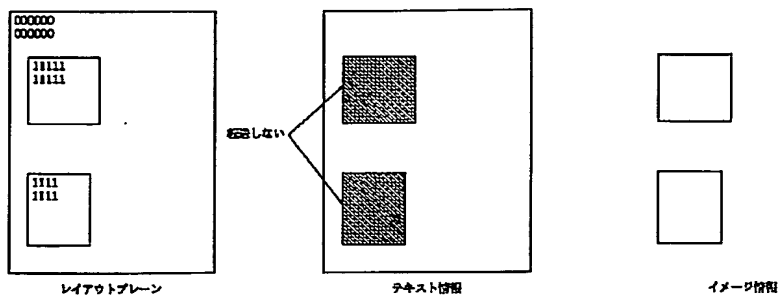
【図17】



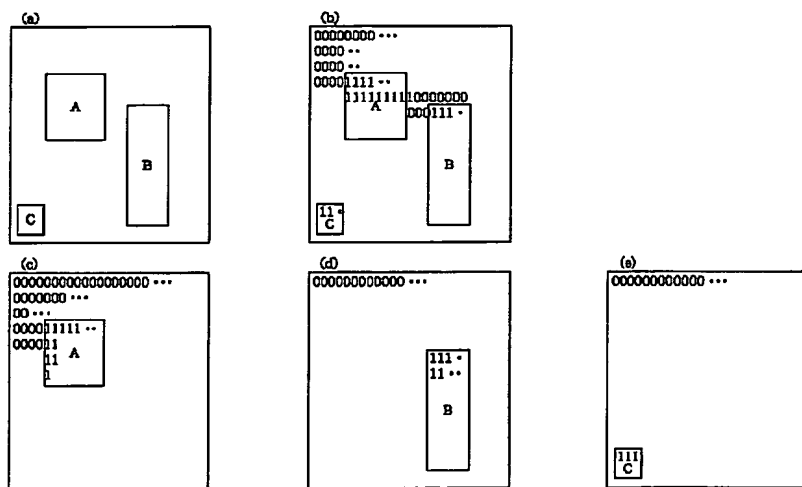
【図19】



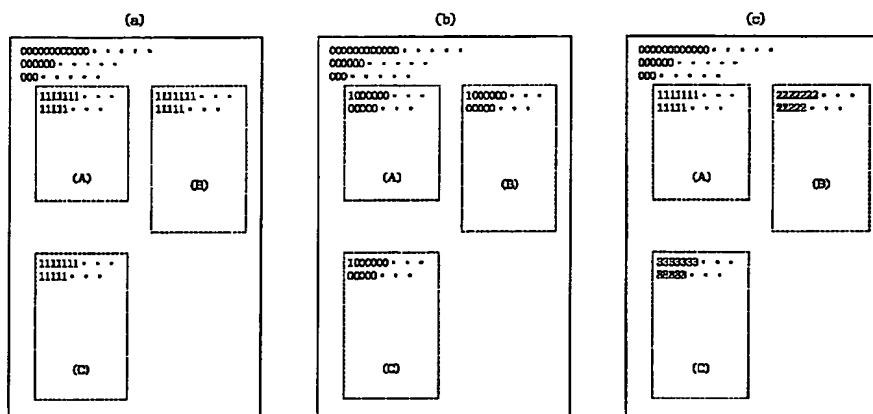
【図 20】



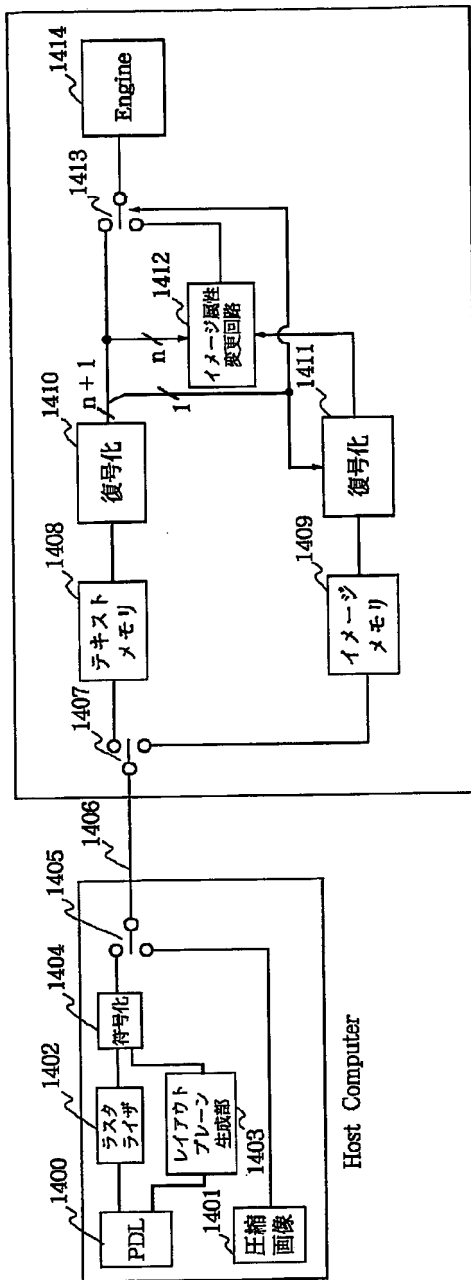
【図 23】



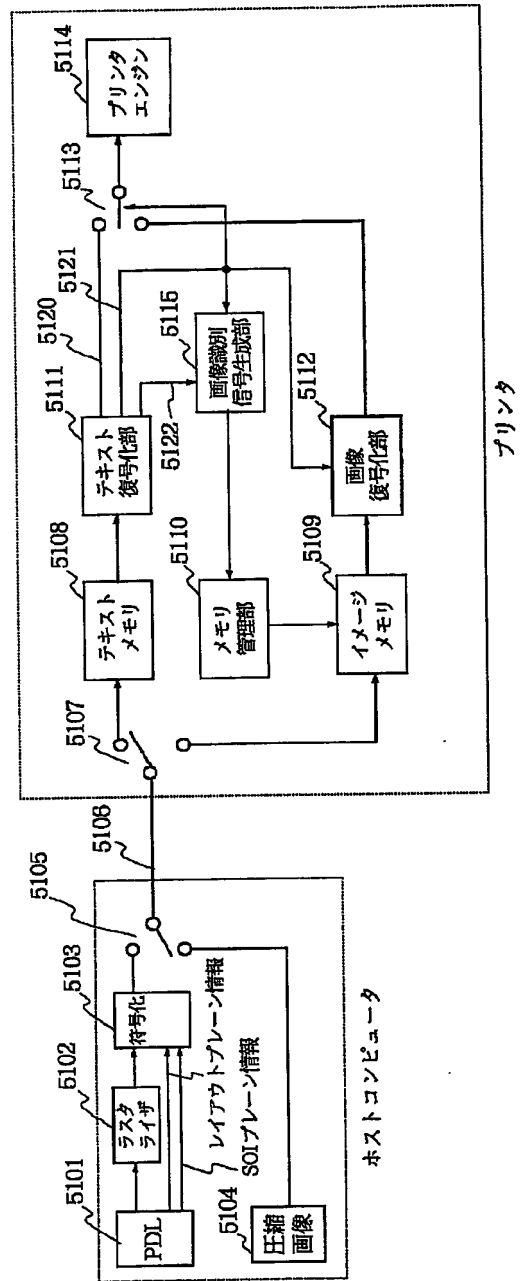
【図 30】



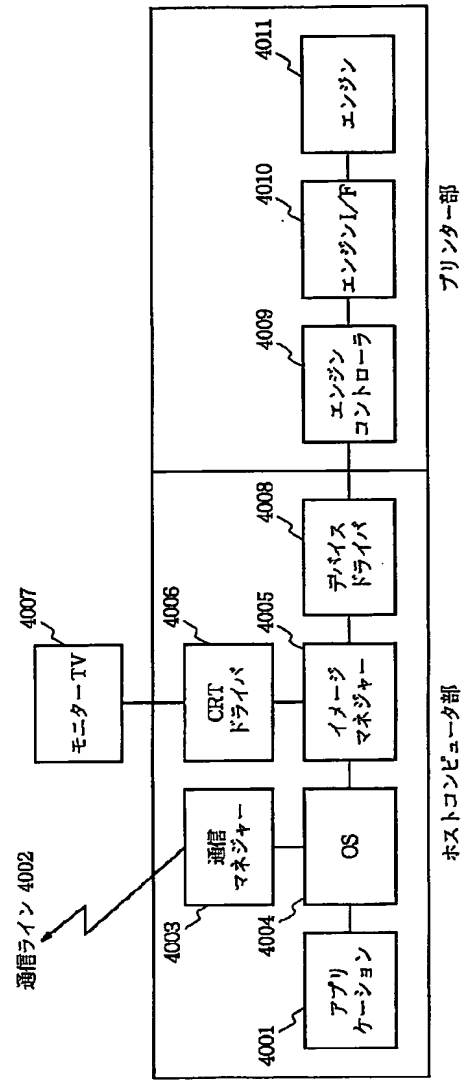
【図 21】



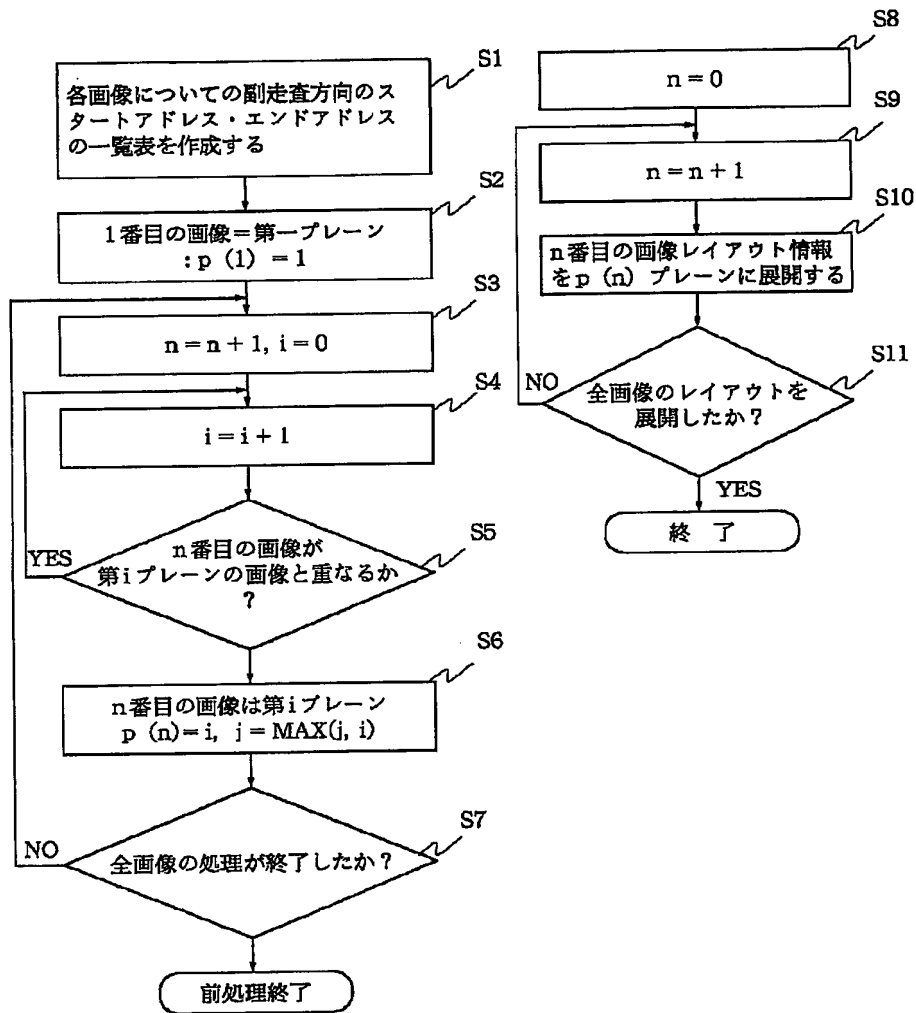
【図 27】



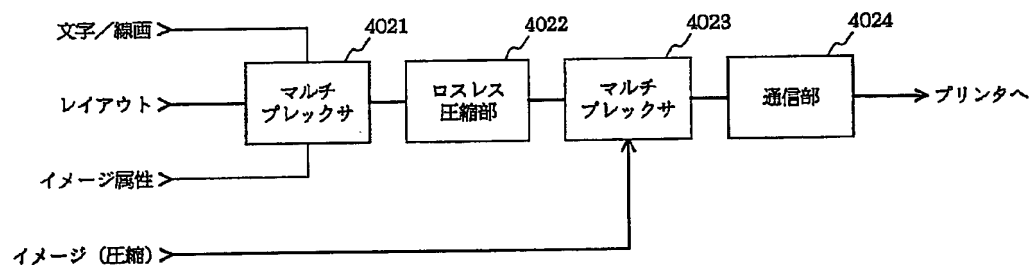
【图 3 2】



【図25】

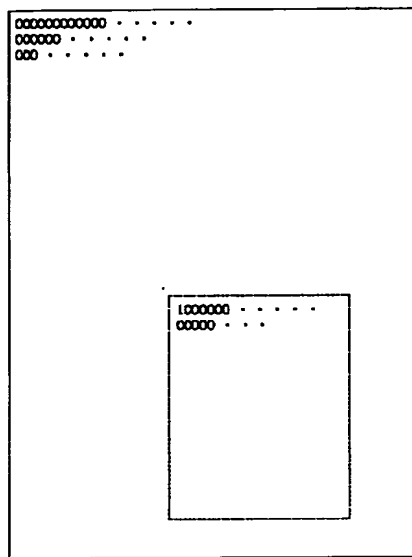


【図34】

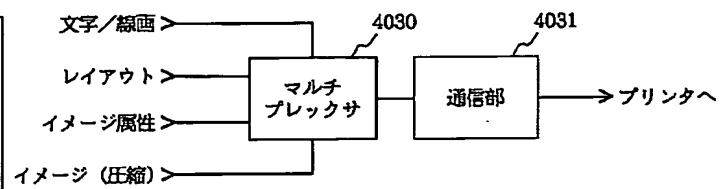


デバイスドライバの機能

【図28】

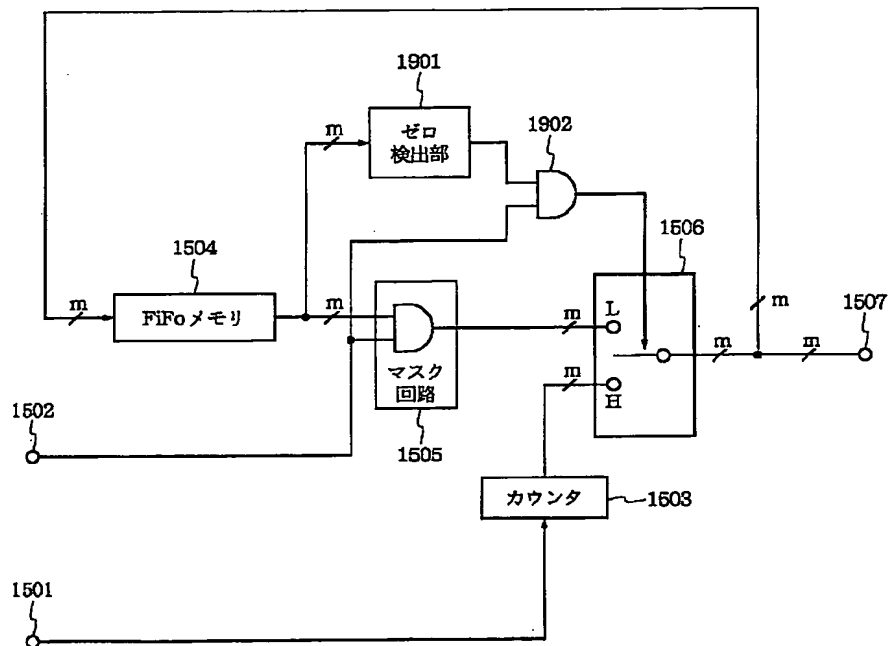


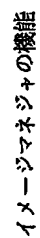
【図36】



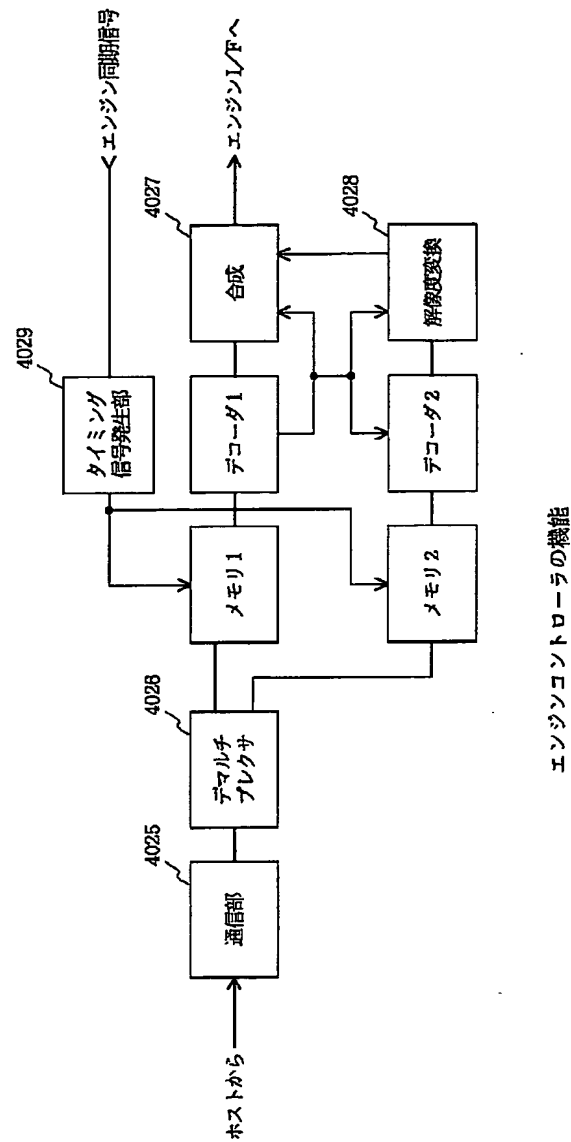
デバイスドライバの第10の実施例

【図31】





【図 35】



エンジンコントローラの機能

フロントページの続き

(51) Int. Cl. ⁵

H 0 4 N 1/41
1/411

識別記号

片内整理番号

F I

技術表示箇所

B 9070-5 C
9070-5 C

(72)発明者 長島 良武
東京都大田区下丸子3丁目30番2号キヤノ
ン株式会社内

(72)発明者 福田 洋美
東京都大田区下丸子3丁目30番2号キヤノ
ン株式会社内